



Handelshögskolan

VID GÖTEBORGS UNIVERSITET

Institutionen för informatik

2005-01-21

Systemförvaltning och fortlöpande systemutveckling

– **”it’s all about version control”**

Abstract

Systemförvaltning är den del av systemutvecklingsprocessen som blir allt mer komplex i samma takt som dagens IT-system blir allt mer komplexa. Komplexiteten med systemförvaltning har ökat till stor del beroende på att det idag allt oftare bedrivs systemförvaltning och fortlöpande systemutveckling. Genom en fallstudie utförd med yrkesverksamma inom systemförvaltning och systemutvecklingsområdet analyserades hur systemförvaltning och fortlöpande systemutveckling bedrivs i en och samma systemmiljö. I uppsatsen framkom ett antal framgångsfaktorer som bör tas i beaktande när systemförvaltning och fortlöpande systemutveckling bedrivs i en och samma systemmiljö. Några av de mest tongivande framgångsfaktorerna är en välfungerande versionshantering, uppdaterad systemdokumentation samt att tydligt utse ägandeskap av systemmiljöer.

Nyckelord: Systemförvaltning, systemutveckling, fortlöpande systemutveckling, parallell systemutveckling, versionshantering.

Författare: Daniel Lindwall
Johan Reventberg

Handledare: Alan B. Carlson
Magisteruppsats, 20 poäng

INNEHÅLLSFÖRTECKNING

Introduktion.....	1
Problemområde	1
Syfte	2
Avgränsning	2
Disposition	3
Målgrupp.....	3
Metod	4
Vetenskapsteoretiskt perspektiv.....	4
Tillvägagångssätt	5
Val av ämne	6
Förstudie	7
Val av undersökningsmetod.....	7
Datainsamlingsmetoder.....	9
Primär information.....	9
Val av respondenter	9
Intervjuernas genomförande	10
Standardisering och strukturering	10
Bearbetning och analys av datamaterial	11
Sammanställning av teorikapitel.....	11
Metodens vetenskaplighet.....	12
Källkritik.....	12
Validitet, reliabilitet och generaliserbarhet.....	13
Teori.....	15
Systemutveckling.....	15
Vattenfallsmodellen	15
Iterativ modell.....	16
Reaktiv modell	17
Systemförvaltning	18
Systemförvaltningsarbetet.....	19
Ändringshantering.....	20
Problem med underhåll av programvaror	21
Dokumentation.....	22
Verktyg för systemförvaltning.....	23
Skillnader systemutveckling och systemförvaltning	25
Systemförvaltning och fortlöpande systemutveckling i en systemmiljö	26
Versionshantering	27

Analys	28
Systemutveckling	28
Systemförvaltning	30
Systemförvaltningsarbetet	30
Ändringshantering	32
Problem med underhåll av programvaror	33
Dokumentation	34
Verktyg för systemförvaltning	34
Systemförvaltning och fortlöpande systemutveckling i en systemmiljö	36
Versionshantering	40
Slutsatser	41
Framtida forskningsfrågor	43
Källförteckning	44
Källor internet	45
Appendix	46
A. Grundläggande begrepp	46
B. Intervjuguide	48

INTRODUKTION

I introduktionskapitlet beskrivs problemområdet med tillhörande syfte. Studiens avgränsning och målgrupp finns även beskrivet.

Problemområde

Förvaltning kan utföras för olika delar i en organisation. Enligt Eklund & Fernlund (1998) är förvaltning av system, dvs. systemförvaltning den del som det läggs en relativt liten vikt vid och resurser på inom en organisation. Systemförvaltning innebär enkelt beskrivet, processen efter det att en programvara har implementerats och underhållsarbete utförs (Nordström & Welander, 2002). Systemförvaltning omfattar åtgärder för att bibehålla eller förstärka ett system (Brandt et al, 1998). Eklund & Fernlund (1998) hävdar att förvaltning av dagens IT-system blir besvärligare i samma takt som dagens IT-system blir allt mer komplexa.

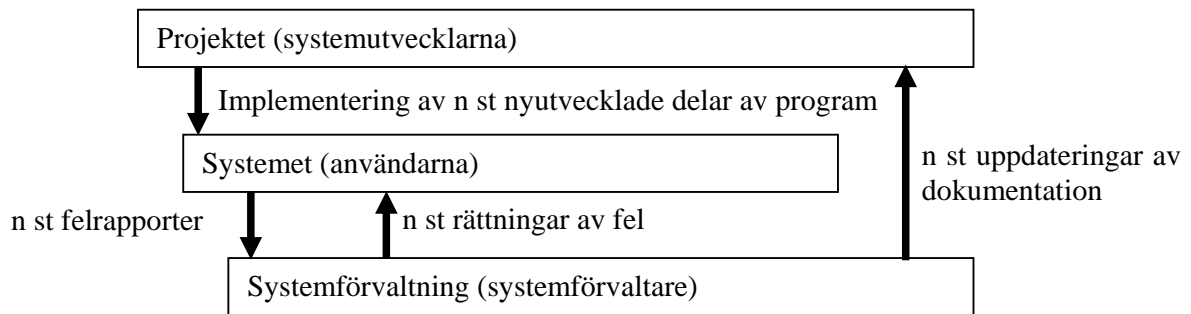
Systemförvaltning har genom tiderna fått ett dåligt anseende för att det har betraktats som synonymt med felkorrigeringar (Eklund & Fernlund, 1998). Felkorrigeringar är bara en liten del av systemförvaltningsprocessen som innehåller ytterligare delar, vilka kommer att beskrivas senare. Begreppet systemförvaltning började dyka upp i början av 60-talet och under 90-talet ökade intresset för systemförvaltning, till stor del beroende på att systemförvaltning tog mycket resurser i anspråk (Brandt, 1992). Siffrorna på hur mycket resurser som togs i anspråk varierar mellan 33 % och 75 % (Nordström & Welander, 2002). Enligt Nordström och Welander (2002) så är den främsta anledningen till att siffrorna varierar så kraftigt att innebörden i systemförvaltningsbegreppet varierar och att olika saker därför mäts.

Eklund & Fernlund (1998) delar in systemförvaltning i fyra delar:

- Korrigering, dvs. åtgärder för lösa ett problem
- Anpassning av programvara till ny maskin- eller programvara
- Förbättring, dvs. utökning av funktionaliteten
- Förberedning av programvaran för kommande utbyggnader

Det som Eklund & Fernlund (1998) beskriver ovan är en förenklad bild som inte överensstämmer med dagens systemutveckling. Dagens systemutveckling är mer dynamisk och flera företag arbetar med en iterativ utvecklingsprocess (Haverblad, 2004). I den iterativa utvecklingsmodellen utförs de olika delfaserna analys, design, implementering och testning (Appendix A) för mindre moduler (Appendix A), till skillnad från den traditionella vattenfallsmodellen där de olika delfaserna avslutas innan de övergår till nästa fas (Andersen, 1994). En anledning och fördel till att fler företag arbetar utifrån den iterativa utvecklingsmodellen är att kunden kan börja använda programvaran tidigare (Budgen, 2003). En följd av den iterativa utvecklingsmodellen är att förvaltningsarbetet börjar vid den första integrationen samtidigt som andra moduler av programvaran fortsätter att utvecklas.

Budgen (2003) hävdar att arbetet med programutveckling samtidigt som systemförvaltning pågår i en systemmiljö (Appendix A) är komplext och problematiskt. När systemutvecklarna har integrerat första versionen kan användarna påbörja användandet av programmet. Användarna hittar vanligtvis ”fel” i programmet som de rapporterar till systemförvaltarna genom felrapporter. Systemförvaltarna åtgärdar felen och uppdaterar en version av programvaran. Systemutvecklarna arbetar samtidigt med att utöka funktionaliteten vilket görs i en annan version av programmet. Ett problem som kan uppkomma är att olika versioner implementeras i systemet hos användarna. Figur 1 ger en överskådlig bild över problematiken inom de olika faserna. Inom parentes anges de olika aktörerna som kan finnas inom de olika faserna.



Figur 1. Överskådlig bild problemområdet (Källa: Författarna, grundas på Révay (1992))

Syfte

Systemförvaltning och fortlöpande systemutveckling är ett område som är relativt outforskat och odokumenterat. Studiens första syfte är att beskriva hur systemförvaltning och fortlöpande systemutveckling bedrivs i en och samma systemmiljö.

Att bedriva systemförvaltning och fortlöpande systemutveckling innebär att många moment ska utföras vilket leder till att flera faktorer bör tas i beaktande. Studiens andra syfte är att finna och beskriva framgångsfaktorer som bör tas i beaktande vid systemförvaltning och fortlöpande systemutveckling i en och samma systemmiljö (Appendix A).

Avgränsning

De informanter som kommer att delta arbetar i olika typer av organisationer och i olika typer av projektstrukturer. Samtliga informanter arbetar med projektbaserad systemutveckling och/eller systemförvaltning i någon form. Vi har inte för avsikt att göra några jämförelser relaterat till projektfaktorer så som projektstorlek, projektstyrning eller motsvarande, då vi anser att det ligger utanför vårt uppsatsområde.

Disposition

Uppsatsen är disponerad enligt följande:

Metod. Metodkapitlet innehåller en beskrivning av hur studien är genomförd och hur data har samlats in samt bearbetas.

Teori. I detta kapitel beskrivs den teori som finns avseende systemförvaltning och systemutveckling. Därefter beskrivs vilka faktorer som spelar roll för systemförvaltning samt fortlöpande systemutveckling i en systemmiljö.

Analys. I analyskapitlet redovisas för de uppgifter som framkommit genom intervjuer. Kapitlet innehåller även en diskussion där empirin relateras till teorikapitlet .

Slutsatser. Här presenteras arbetets slutsatser bland annat gällande de framgångs faktorer som framkommit för systemförvaltning samt fortlöpande systemutveckling.

Appendix. I appendix A finns en förklaring till begrepp som tas upp. Intervjuguiden finns i Appendix B.

Målgrupp

Uppsatsen riktar sig till organisationer och personer som arbetar med systemförvaltning samt fortlöpande systemutveckling. Uppsatsen riktar sig även till studenter som studerar på D-nivå inom ämnet Informatik.

METOD

Kapitlet inleds med att redogöra för denna vetenskapsteoretiska syn. Därefter kommer val av angreppssätt att beskrivas. Därpå redogörs för de datainsamlingsmetoder som nyttjats för undersökningen.

Vetenskapsteoretiskt perspektiv

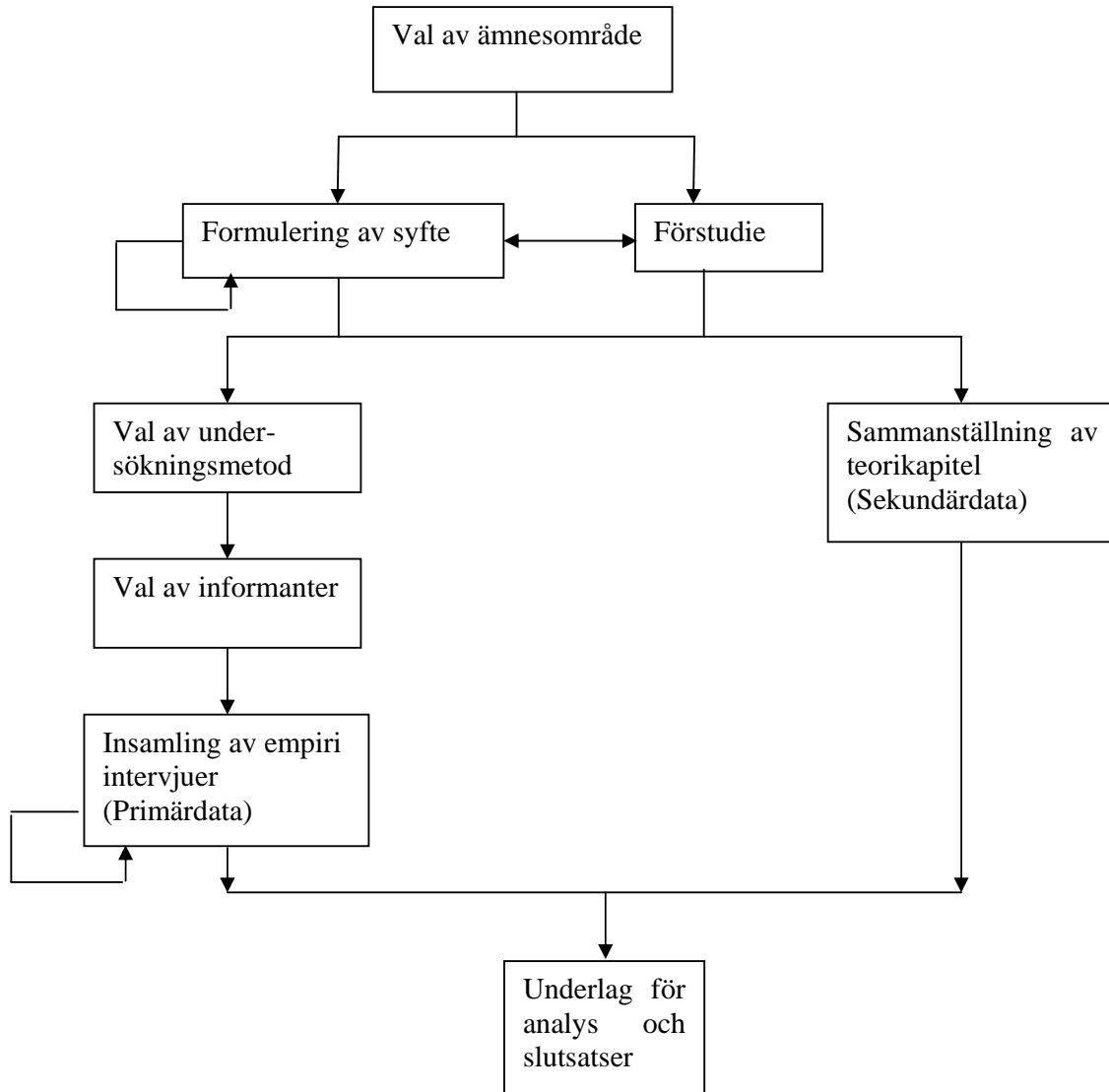
Inom vetenskapsteorin finns två huvudinriktningar, positivism och hermeneutik. Att utgå från ett positivistiskt perspektiv innebär att det finns en absolut sann kunskap (Thurén 1998). Positivismen erkänner två källor till kunskap, iakttagelse och logik. Iakttagelsen sker genom våra fem sinnen, och utifrån denna empiriska kunskap fastställs vad som är fakta. Den logiska sanningen kan enklast förklaras med att två plus två alltid blir fyra. Detta behöver inte empiriskt testas om och om igen. Enligt positivismen finns det en objektiv verklighet som forskaren ska försöka förklara med hjälp av orsak verkan samband och utan egna subjektiva bedömningar (Patel och Davidsson 1994).

Det hermeneutiska perspektivet skiljer sig åt från det positivistiska på flera punkter. Den främsta skillnaden är att en forskare med ett hermeneutiskt angreppssätt tolkar verkligheten utifrån hur han/hon förstår sin omgivning. Denna förståelse bygger på ett igenkännande och empati. Det är alltså forskarens subjektiva uppfattning om verkligheten som avspeglas i hans/hennes forskningsresultat. Om en annan forskare med andra värderingar och preferenser genomför en likadan underökning, är uppenbarligen risken stor att resultatet kommer att avvika från det första fallet (Thurén 1998). Av denna anledning skulle förmodligen en sann positivist förkasta ett hermeneutiskt forskningsresultat.

Att ansluta sig till någon av dessa vetenskapliga inriktningar är ingen lätt uppgift. För den här undersökningen, har vi valt en metod som svarar mot kraven för vetenskaplig sanning utifrån ett hermeneutiskt perspektiv. Avsikten är att söka uppnå en förståelse, inte att förklara. För att uppnå denna förståelse har personliga intervjuer varit ett naturligt val för att få tillgång till information. Det är mycket svårt, och inte heller önskvärt, att bearbeta denna typ av information utan att subjektivitet och egna tolkningar påverkar resultatet.

Tillvägagångssätt

Den grafiska modellen nedan (figur 2) beskriver vårt tillvägagångssätt med uppsatsen.



Figur 2: Tillvägagångssätt (Källa: Författarna)

När vi hade valt vilket ämnesområde vi ville skriva om och fått det godkänt började arbetet med att formulera syftet med uppsatsen. Som en delprocess i framtagandet av syftet utförde vi en förstudie av vårt problemområde. Förstudien gjordes av två anledningar. Den ena anledningen var att skapa en tidig helhetsbild av problematiken runt ämnet. Den andra anledningen var att verifiera syftets relevans. I nästa steg valde vi vilken undersökningsmetod som skulle ligga till grund för vårt fortsatta arbete. De följande stegen i vårt tillvägagångssätt; val av metod och sammanställning av teorikapitel, påbörjades därefter, dock med varierande begynnelsestid.

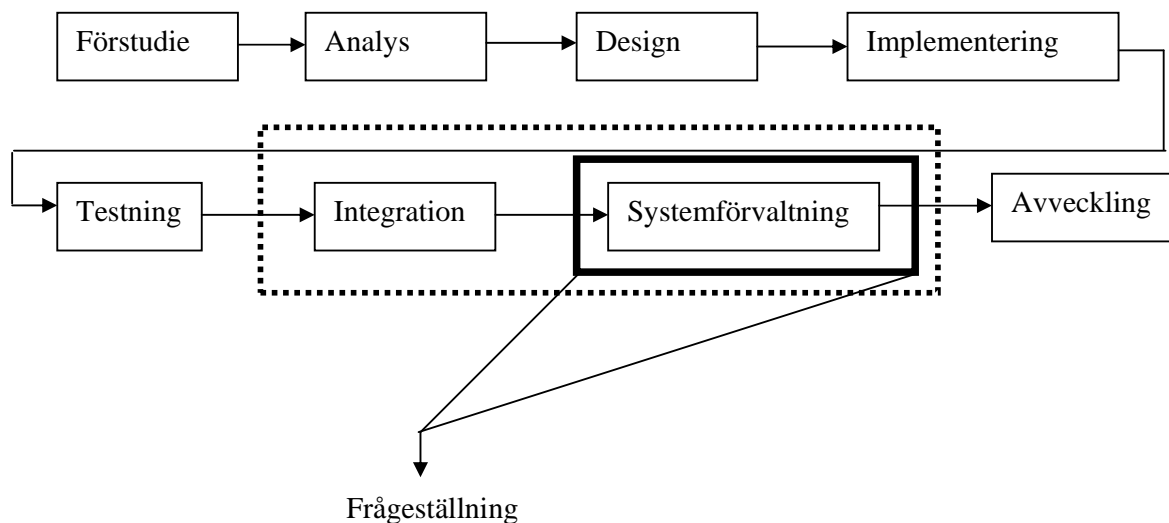
Insamlandet av sekundärdata har skett löpande under i stort sett hela arbetet, medan intervjuerna (primärdata) har skett iterativt och tidsmässigt, under arbetets andra hälft. De ovan nämnda stegen; insamling av sekundärdata och intervjuer är det som utgör förutsättningarna för vår analys och våra slutsatser.

Arbetet med uppsatsen har till mångt och mycket varit en iterativ process, där vi som författare under tidens gång gått tillbaka till de olika delarna/stegen, dels för att erhålla information, men också för att revidera informationen. Det har skett av i huvudsak två anledningar, den ena anledningen är att det under arbetets gång har uppstått händelser som vi varit oförmögna att förutsäga/påverka, t ex informanter som bytt arbetsgivare och därmed avböjt vidare medverkan. Den andra anledningen är att den kunskap som vi som författarna besitter inom ämnet har ökat.

I följande kapitel följer en mer ingående beskrivning av vårt tillvägagångssätt.

Val av ämne

Begreppet systemutveckling omfattar flera olika delar. De kapitel som varit i fokus under arbetet visas i figur 3 nedan.



Figur 3. Systemutvecklingskedjan samt vårt fokus. (Källa: Författarna, inspirerad av Andersen (1994).

Förstudie

För att få en bra bild över problemområdet utfördes en förstudie i två delar. Förstudiens första del bestod av en informell intervju med en IT-konsult med mångårig erfarenhet från stora systemutvecklingsprojekt (Appendix A) samt systemförvaltning. Under intervjun framkom det tydligt att det finns stora utmaningar vid större systemutvecklingsprojekt där systemutveckling och systemförvaltning löper parallellt. Det framkom även att dessa utmaningar potentiellt kan innebära stora kostnader vid införandet av ett IT system.

I förstudiens andra del utfördes litteratursökningar kring problemområdet identifierat i den informella intervjun. Det stod tidigt klart att problemområdet inte var väldokumenterat. Litteratursökningen fokuserades på begreppen systemutveckling och systemförvaltning.

Val av undersökningsmetod

Syftet med metoden i en undersökning är att beskriva för läsaren hur författaren väljer att behandla det aktuella ämnet, genom val av olika vetenskapliga angreppssätt. Holme och Solvang (1997:13) definierar begreppet metod som:

... ett redskap, ett sätt att lösa problem och komma fram till ny kunskap. Allt som kan bidra till att uppnå dessa mål är en metod.

Enligt Patel och Davidsson (1994) kan antingen problemet bestämma metoden, eller också bestäms metoden först. För denna uppsats var det naturligt att låta problemet bestämma tillvägagångssättet, för att i möjligaste mån kunna finna den information som söktes.

Det finns en rad olika metoder för att bedriva forskning. Vilka individer som ska medverka i undersökningen och vilka tekniker som ska användas avgör vilken metod som är lämplig. De vanligaste metoderna är enligt Patel och Davidsson (1994) surveyundersökning och fallstudie (se tabell 1 nedan).

Strategi	Typ av frågor	Kontroll över händelser	Fokusering på samtida händelser
Surveyundersökning	Vem? Vad? Var? Hur många/mycket?	Nej	Ja
Fallstudie	Hur? Varför?	Nej	Ja

Tabell 1: Olika typer av undersökningsmetoder (Källa: Patel & Davidsson, 1994; sid 56).

En surveyundersökning innebär att undersökningen utförs på en större och avgränsad grupp med hjälp av till exempel frågeformulär eller intervju. Surveyundersökningar kan ge information om ett stort antal variabler, men kan också användas för att samla en stor mängd information om ett begränsat antal variabler (Wiedersheim-Paul och Eriksson, 1999).

En fallstudie innebär att undersökningen genomförs på en mindre och avgränsad grupp. Ett fåtal objekt (individer, organisationer etc.) undersöks då ur en mängd perspektiv. Fallstudier utgår från ett helhetsperspektiv och söker skapa en så heltäckande bild som möjligt; på så sätt söker man erhålla en djupare förståelse (Wiedersheim-Paul och Eriksson, 1999). Fallstudier skapar också bättre förutsättningar att komma åt sådan information som kan vara svår att få fram vid en skriftlig enkät eller en kortare intervju. Fallstudien används ofta då forskaren vill studera processer och förändringar (Patel & Davidsson, 1994).

I denna undersökning har fallstudie använts som metod. Enligt Eriksson och Wiedersheim-Paul (1997) kan fallstudier vidare användas i flera sammanhang bland annat som illustration där fallstudien får en förtydligande och pedagogisk funktion eller som hjälpmedel för att skapa hypoteser. Fallstudier kan vara användbara om problemområdet är förhållandevis okänt eller obearbetat, alternativt om en ny infallsvinkel önskas på ett tidigare studerat område (Eriksson & Wiedersheim-Paul, 1997).

Systemutveckling och systemförvaltning har tidigare studerats men då mer separat. Vi har försökt att skapa en ny infallsvinkel genom att göra en koppling mellan systemförvaltning och den systemutveckling som pågår då en version av systemet har implementerats hos kunden. Med hjälp av en fallstudieansats vill vi undersöka vilka problem som finns inom problemområdet. Vi vill undersöka vilka faktorer som är viktiga att tänka på vid systemförvaltning samt fortlöpande systemutveckling. Vi anser att vår empiriska studie faller inom ramen för en fallstudie som ska utföras under 20 veckor. Vår ambition är att komma med generella framgångsfaktorer. I strävan efter en sådan djupare förståelse har vi valt att genomföra intervjuer med personer med koppling till vårt problemområde. Vår intervjuguide återfinns i appendix B.

De personer vi har valt att intervjua jobbar inom olika organisationstyper, t ex linjeorganisation och konsultorganisation. Ett sådant tillvägagångssätt borde belysa problemområdet bättre än en studie av en enskild organisation. Avsikten med vår studie att finna aspekter som bör tas i beaktande och genom att studera flera organisationer kan vi belysa dessa aspekter ur olika synvinklar. Vi menar att detta resonemang styrker valet av fallstudie som metod.

Datainsamlingsmetoder

En kritisk del av fallstudien, och hela arbetsgången, är själva datainsamlingsmetoden, vilken kan grunda sig på såväl primär som sekundär information. Primär information kan inhämtas genom intervjuer, direkt observation och experiment. Sekundär information utgörs av redan insamlat material som litteratur, artiklar, tidigare undersökningar och statistik (Arbnor & Bjerke, 1994).

Primär information

I uppsatsen har intervjuer använts som metod för insamlande av primär information. Ekholm och Fransson (1975) anser att intervjuer ger en möjlighet att utveckla frågeställningar och få ta del av parternas subjektiva åsikter.

När genomförandet av studien hade planerats var nästa steg att besluta utifrån vilka kriterier företag skulle väljas ut. Ambitionen var att undersöka fler än ett företag för att inte få en alltför ensidig bild av problemet. Enligt Eriksson och Wiedersheim-Paul (1997) ökar dessutom möjligheten till jämförelse genom att studera två eller flera organisationer. Samtidigt minskar denna möjlighet då det blir mindre tid till varje studie, det gäller därför att finna en lämplig avvägning (Eriksson & Wiedersheim-Paul, 1997). De utvalda företagen bestod av tre större företag verksamma i Sverige, varav två konsultföretag, som använder sig av systemförvaltning samt fortlöpande systemutveckling. För att få en öppnare dialog gavs möjligheten att hålla informanterna och företag anonyma, vilket de även önskade.

Val av informanter

Enligt Backman (1998) bör personer, eller informanter, väljas ut med hjälp av kriterier som ska kunna ändras under arbetet för att medge ökad förståelse och insikt. För att välja ut lämpliga informanter sattes ett antal kriterier upp. De kriterierna är:

- personerna ska vara verksamma inom ett företag som arbetar med systemförvaltning samt fortlöpande systemutveckling
- personerna ska ha arbetsuppgifter som är direkt relaterade till systemförvaltning och/eller systemutveckling

Informanterna kommer från olika typer av företag. Syftet är att få olika perspektiv på den problematik som finns.

De personer som valts ut är:

Informant A. Arbetar med processutveckling, telekomföretag, Göteborg.

Informant B. Arbetar med kravkonfiguration, telekomföretag, Göteborg.

Informant C. Arbetar som systemförvaltningsstrateg, IT-konsultföretag, Göteborg.

Informant D. Arbetar som projektledare för systemförvaltningsprojekt, IT-konsultföretag, Göteborg.

Informant A och B är från samma telekomföretag. Informant C och D kommer från två olika IT-konsultföretag.

Intervjuernas genomförande

För att få en större förförståelse och en bättre inblick i området gjordes först en genomgång av relevant litteratur inom ämnet. Utifrån tillämplig litteratur utformades sedan intervjufrågorna (se appendix B).

Av de fyra intervjuerna genomfördes samtliga med personerna på respektive företag. Alla intervjuerna genomfördes i avskilda rum, och dessutom med avstängda telefoner och dylikt. Detta resulterade i att intervjumiljön blev så lugn och störningsfri som önskat. Vidare bidrog tidsplaneringen till en lugn och harmonisk intervjusituation. På förhand hade cirka 120 minuter som tid för intervjuerna uppgivits. Detta bedömdes vara väl tilltaget vilket medförde att ingen av intervjuerna gick över tiden. Ett resultat av detta var att det fanns gott om tid till att få alla frågor besvarade på ett tillfredsställande sätt utan att någon tidsbrist uppstod. Fortlöpande under arbetet, efter att de första intervjuerna genomförts, genomfördes kortare telefonintervjuer. Syftet med dessa var att ställa följdfrågor på vad som framkom under de första intervjuerna samt som en följd av att författarnas kunskap i ämnet ökade.

Innan intervjufrågorna började ställas inleddes intervjuerna med en kort presentation av oss, vilket syfte intervjun hade och hur informationen skulle behandlas. Syftet med detta tillvägagångssätt var att klargöra för intervjupersonen vad som efterfrågades. Då bandspelare nyttjades för dokumentation av samtliga intervjusvar begärdes också om tillstånd för detta innan intervjun påbörjades. Bandspelare anses av flera orsaker vara det överlägsna hjälpmedlet vid intervjuer. Användning av bandspelare innebär att ingen information går förlorad då intervjupersonens svar registreras exakt (Patel & Davidsson, 1994). Nyttjandet av bandspelare innebar vidare att koncentrationen var på informanten och på så vis gavs ökad möjlighet till att studera till exempel kroppsspråk, tonfall med mera. Dessa aspekter är att betrakta som en viktig del i den informationsinhämtning som görs med hjälp av intervjuer (Stewart, 1999).

Standardisering och strukturering

Intervjuer kan vara av olika standardiseringsgrad. Vid intervjuer med en hög standardiseringsgrad ställs exakt samma frågor i samma ordning till alla informanter och inga förändringar görs för att frågorna ska passa den aktuella informanten. En låg standardiseringsgrad innebär däremot att intervjuaren har möjlighet att rätta till frågornas formulering och ordningsföljd så att de anpassas till situationen (Andersen, 1994; Patel & Davidsson, 1994).

Inom intervjuteknik talas också om intervjuer med hög och låg grad av strukturering. Graden av strukturering handlar om vilket svarsutrymme informanten ges. En intervju med hög grad av strukturering innebär att svarsalternativen är fasta. En strukturerad intervju lämnar därmed ett mycket litet svarsutrymme för informanten, och intervjuaren kan förutsäga vilka alternativa svar som är möjliga. I en intervju med låg grad av strukturering lämnar frågorna istället maximalt utrymme för intervjupersonen att svara inom (Patel & Davidsson, 1994).

De intervjuer som genomfördes var standardiserade i det avseendet att ett antal frågor ställts till alla informanterna utan att anpassas. Utöver dessa frågor användes ett antal individanpassade/företagsbaserade frågor beroende på företagets organisation och affärsområde. Alla frågor har dock inte ställts i exakt samma ordning och detta beror bland annat på graden av strukturering. Frågorna höll en låg grad av strukturering, det vill säga att intervjupersonerna hade ett stort svarsutrymme. Detta innebar att deras svar ibland kom in på ämnen som inte hade planerats. Ibland gav deras svar också upphov till intressanta följdfrågor och då tillvaratogs möjligheten att ytterligare belysa problemområdet genom att ställa dem.

Av ovanstående resonemang framgår att de intervjuer som genomförts har haft en låg grad av strukturering samt en hybrid av hög och låg grad av standardisering. Syftet med detta tillvägagångssätt var att uppnå en viss standardisering av intervjuerna för att därigenom underlätta en jämförelse mellan informanternas svar. Dock var det intressant att få fram ytterligare, för ämnet, relevant information och intressanta åsikter. För att åstadkomma detta användes ostrukturerade intervjuer och de ovan nämnda individanpassade frågorna.

Bearbetning och analys av datamaterial

Samtliga intervjuer skrevs ut ordagrant. Efter utskriften av intervjuerna lästes de igenom ett flertal gånger och åsikter som ansågs särskilt belysande ströks under, för att eventuellt kunna användas som citat i den färdiga uppsatsen. Intervjувaren strukturerades utifrån ett antal olika områden. Dessa områden var de som vi ansåg utkristalliserades vid intervjutillfällena. De utvalda områdena delades i sin tur in i ett flertal underområden. Analysen tar sin utgångspunkt i det empiriska kapitlet. De empiriska resonemangen varvas med teori samt egna reflektioner och tolkningar.

Sammanställning av teorikapitel

Teorikapitlet har varit svårt att formulera då det finns, som tidigare beskrivet, väldigt begränsat med forskning och därmed litteratur inom detta arbetsområde.

Att ta fram all bakomliggande information som behövs till en undersökning kan i många fall betraktas som omöjligt både ur tids- och kostnadssynpunkt. Detta gör, enligt Yin (1994), sekundärt informationsmaterial attraktivt. Även Andersen (1994) betonar det sekundära materialets nytta i vissa sammanhang. Enligt Andersen är det speciellt lämpat vid planering och förstudier till arbeten som sedan går vidare och använder andra datainsamlingsmetoder. Detta är också något som nyttjats i denna undersökning då skriftlig information om såväl fallföretagen som sponsring ur olika perspektiv varit en viktig del vid planeringen och förberedelserna av studien. Sådant sekundärt informationsmaterial har tillsammans med ett omfattande teoretisk förarbete legat till grund för den intervjuguide som använts vid intervjuerna. Tillvägagångssättet i detta avseende ligger också i linje med Yin (1994) som just påpekar vikten av en teoretisk grund och ett bra förarbete, för att möjliggöra en ändamålsenlig fallstudie.

De sökord vi använt vid litteratursökningen är:

- Systemförvaltning, förvaltning, systemförvaltningsplan
- Systemutveckling, parallell systemutveckling, vattenfallsmodellen, iterativ systemutveckling
- Versionshantering, verktyg för versionshantering
- Application management, software configuration management, software maintenance, software evolution version control

Sökorden har använts dels var för sig men även tillsammans i kombinationer, t ex systemförvaltning + ”iterativ systemutveckling” + versionshantering.

Metodens vetenskaplighet

Enligt Ejvegård (1993) är det tre primära kriterier att tillgodose inom ett vetenskapligt arbete. De tre kriterierna är saklighet, objektivitet och balans. Med saklighet menar Ejvegård (1993) att författaren har ett ansvar för att de uppgifter som han/hon presenterar är kontrollerade. Att kontrollera uppgifterna innebär oftast att författaren går till primärkällan för att bekräfta att de är sanna och riktiga.

Med objektivitet menas att det är viktigt att upptäcka och frigöra sig ifrån sina egna fördomar och förutfattade meningar, i synnerlighet när det föreligger delade meningar inom ett ämne. Författaren måste enligt Ejvegård (1993) alltid sträva efter att återge ståndpunkter från alla viktiga håll. Det är av största vikt att uttrycka om det är ens egna åsikter eller om det är någon annans åsikter. När åsikter kommer från någon annan är det författaren som måste analysera källan och dra en slutsats om källan är objektiv eller inte (Ejvegård, 1993). Saklighet och objektivitet kan enligt Ejvegård (1993) uppnås genom källkritik.

Att sträva efter balans i en uppsats innebär först att en avvägning av det material som författaren har görs, och sedan att kunna publicera det i rätt proportioner i förhållande till dess väsentlighet (Ejvegård, 1993).

Källkritik

Med källkritik menas att författaren tar ställning till en källas trovärdighet. Trovärdigheten hos en källa kan bedömas med hjälp av en inre och en yttre analys (Holme och Solvang, 1997). Den yttre analysen innebär att författaren jämför information mellan olika källor som är oberoende av varandra. Vid en hög trovärdighet ska informationen stämma överens. Vid en inre källanalys bör ställning tas till faktorer som upphovsmannens subjektiva perspektiv och den generella säkerheten i källans innehåll (Holme och Solvang, 1997). Den inre analysmetoden är lämplig för den primär- och sekundärdata som författaren använder sig av.

Det finns även andra aspekter utöver de yttre och inre, att ta ställning till vid en analys av trovärdigheten rörande primär- och sekundärdata. En av de viktigaste och mest komplexa aspekterna som påverkar trovärdigheten är den så kallade intervjuareffekten

(Eriksson och Wiedersheim-Paul, 1999). Det viktiga med intervjuareffekten är att känna till att det alltid uppstår någon form av samspel mellan intervjuare och den som blir intervjuad och att detta kan påverka resultaten på ett icke önskvärt sätt. Intervjuareffekten innebär således att intervjuaren kan påverka informanten genom viss riktad frågeställning, följdfrågor etc., så att svaret blir som intervjuaren vill ha det. För att undvika intervjuareffekten har vi i förväg informerat informanterna om vilka områden och typ av frågor vi kommer att ställa. Det ger honom/henne möjligheten att fundera över frågorna och komma med så korrekta svar som möjligt. Vad gäller de uppföljande telefonintervjuerna har vi inte haft möjlighet att skicka frågorna i förväg vilket negativt kan ha påverkat intervjuareffekten.

Vi har i största möjligaste mån tagit ställning till trovärdigheten hos de källor som vi använt oss av. Vi har, under arbetet med att studera och sammanställa sekundärdata, strävat efter att använda oss av så många källor som möjligt. Vi har också tagit stor hänsyn till upphovsmannens subjektiva perspektiv, och där varit hårda i vår bedömning av källors objektivitet. Avseende de intervjuer som vi utfört, är det möjligt att det förekommer ett visst mått av subjektivitet hos informanterna eftersom de gärna talar väl för den egna organisationens metoder och lösningar. Vi har i övrigt valt att lita på källor som i sin tur refererar till andra källor, med andra ord har vi inte gått till ursprungskällan eftersom vi varit starkt tidsbegränsade.

Validitet, reliabilitet och generaliserbarhet

Tre viktiga variabler vid en undersökning är validitet, reliabilitet och generaliserbarhet.

Enligt Wallén (1996) innebär validitet att enbart det som avses att mäta utförs det mätning på. Reliabilitet innebär att resultatet som författaren kommer fram till genom olika mätningar på samma objekt, ger lika värden. Validitet och reliabilitet är mått på kvalitén i en mätning. De står i ett beroendeförhållande till varandra och det går således inte att fokusera på endast en av de båda faktorerna. För att nå en genomgående hög kvalitet på mätningen måste därför en samverkansbalans mellan de båda faktorerna, validitet och reliabilitet, åstadkommas (Patel och Davidsson, 1991).

Patel och Davidsson (1991:85) konstaterar förhållandet mellan validitet och reliabilitet genom att konstruera ett antal regler:

- Hög reliabilitet är ingen garanti för hög validitet.
- Låg reliabilitet ger låg validitet.
- Fullständig reliabilitet är en förutsättning för fullständig validitet.

Enligt Wallén (1996) kan validiteten definieras genom en teoretisk fråga. Det innebär att det som avses att mäta är klart definierat och avgränsat samt att sambandet mellan variablerna i teorin och det som kan mätas är kartlagt. Validitet kan delas in i två aspekter; inre och yttre validitet. Inre validitet innebär att de begrepp och de mätbara definitionerna av begreppen ska stämma överens. För att kunna undersöka den inre validiteten ska författaren inte behöva samla in empiriska data. Med yttre validitet avses överensstämmelsen mellan verkligheten och det mätvärde författaren får när han/hon använder en mätbar definition. Författaren kan inte bedöma den yttre

validiteten utan att veta hur det empiriska materialet har samlats in och ser ut (Eriksson och Wiedersheim-Paul, 1997). Validiteten i vår undersökning anser vi vara hög eftersom vi har undersökt det som avsetts att mätas, dvs. systemutveckling, systemförvaltning samt de aspekter som ska tas i beaktande när de utförs parallellt. Validitet har uppnåtts genom förstudier och pilotstudier vilka gav oss pålitliga intervjuguider.

Prövning av reliabilitet kan enligt Holme och Solvang (1997) utföras genom att mätningen upprepas. En hög reliabilitet finns om skilda och oberoende mätningar ger samma eller ungefärligen samma resultat. Vi anser att vi har uppnått reliabilitet i vår studie eftersom vi genomfört flera intervjuer med informanter som har olika bakgrund och infallsvinklar. De olika intervjuerna har gett liknande svar vilket tyder på en hög reliabilitet.

Generaliserbarhet innebär hur användbar och allmängiltig den kunskap är som framkommit (Wallén, 1996). Det här blir extra intressant när en fallstudie utförs som en undersökning av en specifik företeelse. Wallén (1996) påvisar att val av fall är viktigt när bedömning av resultatens generaliserbarhet görs, samt när ett ställningstagande görs till om det som undersöks är representativt eller vanligt förekommande. Eftersom vi utfört en fallstudie inom en specifik bransch och för några specifika företag anser vi att generaliserbarheten är medelhög. Det problemområde vi har undersökt är relevant för flera företag inom branschen. Med den motiveringen anser vi att vårt resultat är relativt generaliserbart.

TEORI

I teorikapitlet beskrivs relevanta teorier från det valda problemområdet som ligger till grund för uppsatsen. Kapitlet inleds med en diskussion samt beskrivning av begreppet systemutveckling och den andra delen diskuterar och beskriver begreppet systemförvaltning. Det tredje och avslutande kapitlet behandlar de två begreppen tillsammans, dvs. när systemförvaltning utförs samtidigt som systemutveckling fortskrider.

Systemutveckling

Sommerville (2004) definierar begreppet systemutveckling som ett samlingsnamn av alla aspekter från de tidiga stadierna av systemspecifikation till påföljande underhåll av ett informationssystem. Pressman (2000) definierar däremot begreppet systemutveckling som en process bestående av ett par väl definierade process steg, vilka avverkas med stöd av en samling av olika metoder och verktyg.

Systemutveckling är som tidigare beskrivet ett stort begrepp som innehåller flera olika delar. Några exempel på traditionella strukturerade metoder är Structured Analysis and Design Techniques (SADT) (Ross 1985), Specification and Description Language (SDL) (Belina & Hogrefe, 1989) och Jackson System Development (JSD) (Jackson 1983).

Budgen (2003) beskriver tre olika systemutvecklingsmodeller; den linjära, även benämnd som vattenfallsmodellen, den inkrementella samt den reaktiva.

Vattenfallsmodellen

I den linjära vattenfallsmodellen utför en fas eller process under en viss tidsperiod. När den fasen eller processen är klar påbörjas nästa fas. De faser som ingår i vattenfallsmodellen är (Budgen, 2003), vilka även åskådliggörs i figur 4 nedan.

Förstudie

Syftet med förstudien är att utforska behovet av systemet samt ge vissa riktlinjer för det fortsatta arbetet.

Analys

Här är syftet att identifiera vad användaren (Appendix A) behöver, dvs. vad slutanvändaren vill kunna göra med det slutliga systemet.

Design

Detaljerat besluta hur systemet ska konstrueras för att tillgodose den kravspecifikation (Appendix A) som finns. Mer specifikt innefattar designfasen datakrav dvs. vilken information ska systemet hantera samt hur ska denna information hanteras. Vidare innefattar den hur systemet ska byggas/utvecklas samt hur ska systemet se ut.

Implementering

Implementering innefattas av kodning av den detaljerade designen.

Testning

Fasen innefattar testning av olika enheter, t ex enhet och moduler (Appendix A), samt slutligen integration av systemet till användarna. (Shelly et al, 2002)

Integration

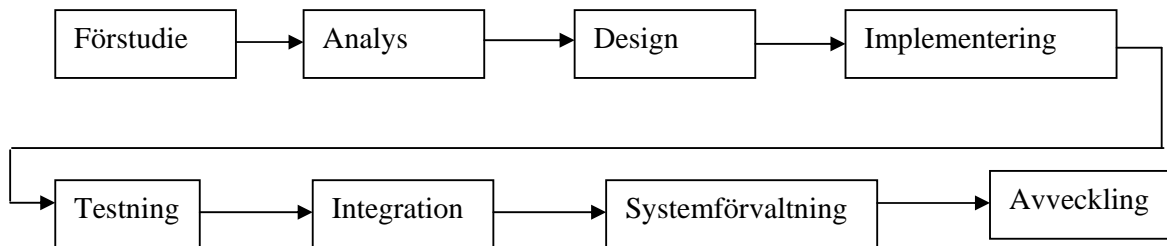
Integration avser den aktiviteten då den utvecklade programvaran integreras i kundernas systemmiljö.

Systemförvaltning

Fasen systemförvaltning innefattar flera olika delar. Det involverar bl a förändring och förbättring. Vi kommer att beskriva förvaltning senare i teorikapitlet.

Avveckling

Avveckling är den sista fasen vilket innebär att ett system tas ur bruk



Figur 4: Systemutvecklingsmodellen, vattenfallsmodellen (Källa: Författarna, inspirerad av Andersen (1994), samt Budgen (2003))

Iterativ modell

En iterativ eller inkrementell utvecklingsmodell baseras på de grundläggande faserna i vattenfallsmodellen beskrivna ovan. En iterativ utvecklingsmetod kan förklaras som en minivattenfallsmodell där varje steg existerar fast är betydligt mindre. Karaktäristiskt för en iterativ utvecklingsmetod är att genererar snabbare resultat med mindre grundläggande information och erbjuder större flexibilitet. (Shelly et al, 2002). En aspekt med att använda den iterativa systemutvecklingsmodellen är att användarna tidigare kan få tillgång till en del av systemet. Det kan innebära fördelar, som t ex att användarna kan få tillgång till delar av systemet tidigt vilket kan innebära att förändringar kan genomföras för att användarna ska bli nöjda (Budgen, 2003). Användning av olika typer av prototyper (Appendix A) är vanligt (Budgen, 2003).

Det positiva med den iterativa modellen är att små processer ger bättre effektivitet vilket leder till att systemförvaltningen blir så lättarbetad som möjligt (Shelly et al, 2002).

Andersen (1994) hävdar att systemutveckling utifrån den iterativa modellen kan se olika ut med avseende på antal iterationer. Det finns inga riktlinjer för hur många iterationer som bör finnas i ett systemutvecklingsprojekt utan det beror mer på storlek, tid och resurser (Sommerville, 2004).

Reaktiv modell

En reaktiv systemutvecklingsmodell kan ses som en variation till den iterativa modellen. Utveckling av 'open source' kod är ett bra exempel på reaktiv systemutveckling. Denna typ av utveckling skiljer sig mycket från de tidigare beskrivna metoderna, fast det är en form av iterativ utveckling (Budgen, 2003). De stora skillnaderna ligger mer i organisationen än i själva modellen. Istället för att samla idéer från ett litet antal beslutsfattare involverar open source modellen ett antal koordinatörer som samlar idéer från många intressenter (Budgen, 2003). En markant skillnad mot den iterativa modellen är att den reaktiva modellen inte har någon planering.

Enligt Reváy (1992) utförs systemutveckling ofta i projektform. Reváy (1992) anser att för att kunna utföra ett effektivt projektarbete samt få ett effektivt styrinstrument bör systemutvecklingsmodellen väljas med noggrannhet.

Systemförvaltning

Vad är systemförvaltning egentligen? Frågan är inte helt okomplicerad att besvara. Inom "systemförvaltningsområdet" finns en flora av begrepp som används för att beskriva snarlika, och ibland samma, saker. Exempel på begrepp för att beskriva systemförvaltning är, systemunderhåll, software maintenance, re-development, forward engineering, application management, och software evolution för att bara nämna några (Stenberg, 1993). Dessutom är gränsen mellan systemutveckling, vidareutveckling och systemförvaltning väldigt diffus - beroende på vem som tillfrågas så erhålls olika svar. En definition - som är relevant för den fortsatta diskussionen - ges av M. M. Lehman (Brandt, 1992)

För mjukvara används termen förvaltning vanligen för att beskriva alla förändringar man gör i ett program efter dess första installation.

Definitionen är relativt generell om den jämförs med andra definitioner, men den innefattar inte alls sådana aktiviteter som användarstöd, utbildning eller drift (Nordström, 1993). Detta gör dock inte definitionen mindre användbar då syftet är att undersöka objektorienteringens inverkan på förvaltningsprocessen.

En annan definition ges av Blackburn (2000):

Systemförvaltning är det totalt antal aktiviteter som krävs för att ge kostnadseffektiv support till ett mjukvarusystem.

Då referensgruppen för Systemförvaltning (RFS) startades 1990 var deras första uppgift att utveckla en definition för begreppet systemförvaltning. Deras definition från 1991 är:

Systemförvaltning är samtliga aktiviteter som görs för att administrera och hantera ett informationssystem i drift, så att det under hela dess livstid effektivt bidrar till att uppfylla verksamhetens mål

Vad är det som avses med förvaltning och vad är det som ska förvaltas. Enligt Nordström & Welander (2002) finns det flera olika definitioner på det som förvaltas. De använder t ex informationssystem och avser då teknisk plattform, applikationer och verksamhet. Begreppet är dock vagt och det är diffust vad som avses. Därför har begreppet förvaltningsobjekt börjat användas. Begreppet förvaltningsobjekt är mer användbart då det kan användas för att beskriva förvaltning av flera olika typer av andra objekt som processer och helpdesk-funktioner.

Systemförvaltningsarbetet

Enligt Révay (1992) är en föråldrad syn på systemförvaltning att det bedrivs i linjeverksamhet. En ny syn på systemförvaltningsarbetet är att det ska utföras i projektarbete. Ett projekt ska enligt Révay (1992) ha en styrgrupp, projektledning samt specialister i berörda arbetsgrupper och referensgrupper. Projekt blir en klart avgränsad aktivitet med tydlig identitet. På samma sätt ska systemförvaltning kunna karakteriseras. Den ska vara avgränsad och tydlig. Det ska finnas avsatta resurser med ledning samt tillgång till specialister. Projekt skapas för att lösa problem och när problemet är löst upphör projektet. Projektperspektivet är därför lämpligt att tillämpa i samband med den nya synen på systemförvaltning.

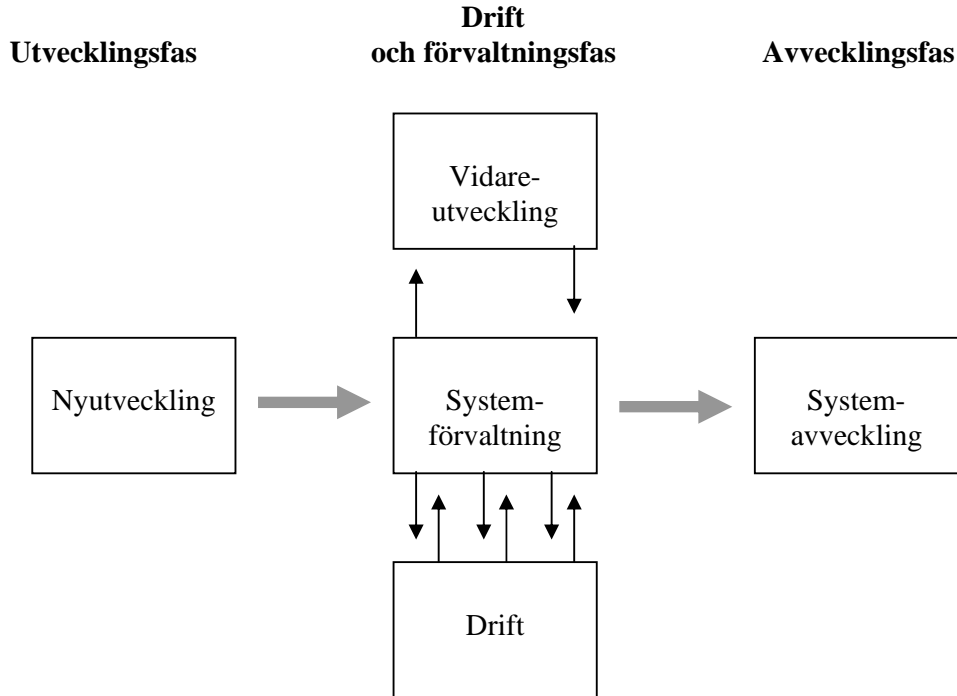
För att kunna utföra denna förändring av synen på förvaltning anser Révay (1992) att följande bör vara fastlagt:

- Systemförvaltningsarbetet ska vara jämställt med nyutvecklingsarbete beträffande ledning, beslutsförmåga och arbetsförhållanden.
- Serviceanda i systemförvaltningsverksamheten.
- Systemförvaltningsdebatten ska vara aktuell avseende nyheter på området vad gäller hjälpmedel, metoder eller verktyg.
- Någon form av belöningssystem som styr mot målen och tillhandahåller ett antal belöningar för att uppmärksamma enskilda prestationer.

I boken "Modern systemförvaltning" beskriver Révay (1992) systemets livscykel utifrån tre faser:

1. Utvecklingsfasen, dvs. utvecklingen från problemdefinition till systemimplementering.
2. Drift- och förvaltningsfasen, dvs. den tidsperiod då systemet är i drift.
 - a. Driften omfattar de åtgärder som erfordras för att applikationen i samverkan med systemprogrammen och hårdvaran ska kunna producera utdata på avsett sätt.
 - b. Systemförvaltning omfattar de åtgärder som erfordras för att administrera, styra och hantera ett system som är i drift.
 - c. Vidareutveckling avser större förändringar av systemet och utgör ett komplement till systemförvaltningen.
3. Avvecklingsfas avser åtgärder för att avveckla ett system.

Figur 5 nedan ger en grafisk representation.



Figur 5. Systemets livscykel (Källa: Révay, 1992)

Nordström & Welander (2002) har delat in förvaltningsarbetet i tre aktiviteter vilka beskrivs nedan:

1. Förvaltningsstyrning, vilket den aktivitet som har som syfte att administrera och styra förvaltningsarbetet.
2. Användarstöd, vilket avser aktiviteter för att ge slutanvändarna stöd i problemsituationer samt aktiviteter för att förebygga problem
3. Ändringshantering, vilket avser aktiviteter från det att ett ändringsbehov uppstår till dess ändringen är införd. Enligt Nordström & Welander (2002) är ändringshantering den mest frekvent förekommande systemförvaltningsåtgärden och den beskrivs ytterligare nedan.

Ändringshantering

Ändringshantering är den aktivitet som har betraktats som synonymt med förvaltningsarbete (Nordström & Welander, 2002). Det finns flera bakomliggande orsaker till ett ändringsbehov (Appendix A) och här är några exempel (Nordström & Welander, 2002):

- Upptäckt av fel i någon del av systemet.
- Upptäckt av brister i någon del av systemet
- Förändringar i verksamheten avseende
 - Regelverk
 - Organisation
 - Produkter och tjänster
- Förändringar i IT-stödet
 - Applikation
 - Teknisk plattform
 - Förändringar i relaterade objekt

Nordström & Welander (2002) har utfört en undersökning och funnit fyra kategorier av ändringar som förekommer i förvaltningsarbetet (RAFS):

- rättning
- anpassning
- förbättring
- sanering

Nordström & Welander (2002) har utfört en studie av 20 förvaltningsplaner från affärsdrivande organisationer, myndigheter, kommuner och landsting. Resultatet visar att förbättring utgör 50 % av förvaltningsarbetet. Hela tabellen kan ses nedan (tabell 2).

Kategori	Andel i %
Rättning	19
Anpassning	27
Förbättring	50
Sanering	4

Tabell 2: Fördelning av andel av arbete (Källa: Nordström och Welander, 2002)

Problem med underhåll av programvaror

Ett stort problem med underhåll av programvaror är att programvaror inte utvecklas enligt en definierad mall eller enligt konstens regler. När programutvecklingen startar så finns förhoppningen att bygga en strukturerad och underhållsvänlig programvara men på grund av resurs- eller tidsbrist görs vissa inskränkningar. En av anledningarna som Eklund & Fernlund (1998) anger är att förvaltning har låg status för programleverantörerna.

Det finns flera olika svårigheter med förvaltning och dokumentation bidrar till en stor del av den svårigheten (Eklund & Fernlund, 1998). Dokumentation kan i flera fall vara undermålig, felaktig eller borta (Eklund & Fernlund, 1998). Programmet kan även ha genomgått flera förändringar vilket kan leda till att dokumentationen inte längre stämmer överens med programvara.

Ett annat problem som kan försvåra underhåll av programvara kan vara den miljö som programvara verkar i. Exempel på miljö är t ex maskinvara, operativsystem, samarbetande applikationer, nätverk etc. Den miljön kan skilja sig från den miljö där programmet skapades. Det som kan skilja är t ex programspråk, debugger, kodningsstandard, dokumentstandard, dokumentationsverktyg etc.

Dokumentation

Dokumentation kan som tidigare beskrivits försvåra underhåll av programvara. Finns det en dokumentation av programvaran bör den uppdateras så att den stämmer överens med de nya förutsättningarna. För att ha koll över de olika åtgärderna som utförs bör en underhållsrapport skrivas. Enligt Eklund & Fernlund (1998) bör underhållsrapporten minst innehålla:

- Programinformation (namn på programmet, version, typ av plattform, språk, storlek etc.).
- Installationsdatum hos användaren.
- Antal rapporterade fel sedan installationen.
- Hänvisning till underhållsförslaget för att kontrollera att åtgärder har utförts på det som efterfrågades.
- Typ av underhåll, eventuellt med gradering av hur allvarligt felet var.
- Underhållsdatum, start- och slutdatum.
- Omfattning av underhållet (Rader som lagts till eller tagits bort, antal mantimmar, vilka resurser som användes etc.).
- Förändringar av dokumentationen.
- Förtjänster av underhållet. Vilka felkorrigeringar, förbättringar, anpassningar etc. har utförts.
- Version på det nya programmet (stor eller liten versionsutveckling).

Vid förändring av funktionaliteten i programvara kan det ge upphov till bieffekter som i sin tur kan leda till fel. Då programvaror är komplexa kan förändring i kod leda till introducering av nya fel.

Eklund & Fernlund (1998) har tre kategoriseringar som kopplas till sidoeffekter:

- Kod. Delar av programmet har döpts om eller tagits bort och ändringar kan medföra sidoeffekter. Enklare sidoeffekter kan upptäckas under kompilering men stora fel kan förbises och upptäckas av användarna.
- Data. Förändringar i datastruktur och databeroende, t ex ändringar av konstanter, filformat, länkade datastrukturer eller storlek på fält, kan leda till sidoeffekter
- Programmets dokumentation. Förändringar av programmet medför att dokumentationen blir inaktuell och utförs inte uppdatering kan det medföra sidoeffekter senare i vidareutvecklingen.

Verktyg för systemförvaltning

För att kunna hantera systemförvaltningen på ett effektivt sätt finns ett antal standardiserade program för att underlätta systemförvaltningsarbetet. De huvudområden inom systemförvaltningen som kan effektivteras och underlättas av ett systemstöd är:

- Versionshantering
- Felrapportshantering

De olika programmen som finns på marknaden skiljer sig en del åt men delar samtliga oftast samma grundprincip. Bland de största aktörerna på marknaden är IBM – Rational, Computer Associates – AllFusion/Harvest, Microsoft - Visual Source Safe. Dessa leverantörer tillhandahåller samtliga en mängd olika verktyg för stöd av hela eller olika delar av systemutvecklingsprocessen.

Versionshantering

En programvara som används för versionshantering är Visual Source Safe (Microsoft, MSDN). Visual Source Safe följer samma trädstruktur princip som Microsoft's filhanterare. Användaren skapar en trädstruktur med mappar där filerna sparas och behandlas. Programmet ger användaren möjlighet att hämta en fil och arbeta med den. För att ingen annan ska kunna arbeta med den måste användaren markera filen som upptagen under tiden som han/hon arbetar med den. När ändringen/uppdateringen är klar laddar användaren upp den nya versionen av filen och gör den tillgänglig för andra användare. Verktuget innefattar också ett stöd för att automatiskt slå samman en eller flera versioner av samma kod. Verktuget jämför de olika versionerna av en viss kod och visar visuellt för användaren var i koden de olika versionerna skiljer sig åt. Verktuget ger därefter användaren möjlighet att välja ett av de alternativ som visas. På det sättet kan man halvautomatiskt slå samman en version med en annan.

Harvest (Computer Associates) omfattar motsvarande funktionalitet som Visual Source Safe fast med en lite annorlunda struktur. I Harvest arbetar man med att filerna som behandlas ligger i samma struktur som de gör i den miljö där de används. Det innebär att filerna (koden) som behandlas ligger i samma struktur som den ska göra när den implementeras i en produktionsmiljö. För varje förändring/felrättning skapar användarna så kallade paket som de sedan associerar till den plats där filen som ska rättas ligger. Filerna checkas in och ut på motsvarande sätt som i Visual Source Safe.

Ett annat program för versionshantering är ClearCase (IBM Rational (1)). ClearCase har liknande funktionalitet som Visual Source Safe.

Felrapportshantering

Rational tillhandahåller en programvara för felrapportering som heter ClearQuest (IBM Rational (2)). ClearQuest ger användaren möjlighet att rapportera fel, dvs. skapa felrapporter. Dessa felrapporter kan sedan hanteras i verktyget från det att den inrapporterats till det att de är avklarade. Olika användare kan hantera och behandla en felrapport som är skapad i ClearQuest. Verktyget kan användas för att hantera prioritering, logga status, kontrollera att det inte förkommer någon konflikt (dvs. det finns mer än en felrapport som behandlar samma problem) tilldelning av felrapporten till olika individer osv. (IBM Rational (2)).

Det finns en mängd program som tillhandahåller motsvarande funktionalitet som ClearQuest, beskrivet ovan. Vad som är gemensamt för samtliga är att alla bygger på en databas med ett enkelt användargränssnitt.

Skillnader systemutveckling och systemförvaltning

I en rapport från Riksdataförbundet (1999) beskrivs skillnaderna mellan systemutveckling och systemförvaltning. Ett utdrag från rapporten ses nedan i tabell 3.

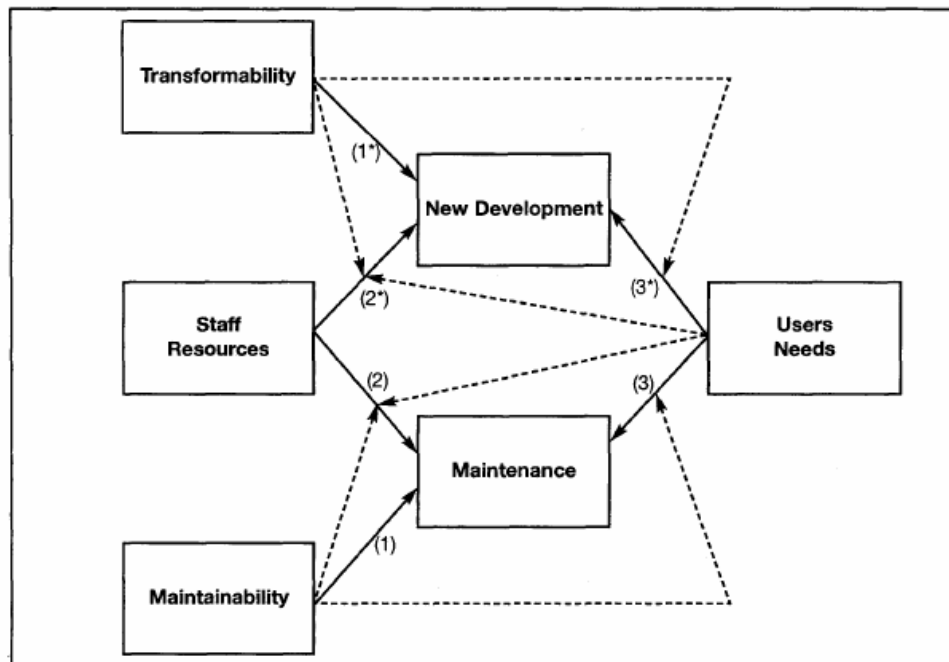
	Systemutveckling	Systemförvaltning
Typ av verksamhet	Temporär	Kontinuerlig
Organisation	Projekt	Linje
Ledarskap	Entreprenör	Koordinator
Styrning	Slutmål	Behov/efterfrågan
Aktiviteter	Skapande	Vårdande
Resurser	39 %	52 %
Litteratur	250	1

Tabell 3. Skillnader mellan systemutveckling samt systemförvaltning (Källa: Riksdataförbundet 1999)

Tabell 3 ovan beskriver skillnaderna mellan systemutveckling och systemförvaltning. Resultatet är inte absolut utan Riksdataförbundet har presenterat ett generellt resultat. Ur tabellen kan utläsas att systemutveckling bedrivs i temporär verksamhet, vanligtvis projekt, medan systemförvaltning bedrivs i en kontinuerlig verksamhet, vanligen kallad "linje". Ledarskapet skiljer sig åt genom att systemutveckling leds av en entreprenör som strävar efter ett slutmål. Systemförvaltningsarbetet leds av en koordinator som styrs av behov och efterfrågan. Aktiviteterna är skapande inom utvecklingsarbetet men vårdande i förvaltningsarbetet. Resurserna varierar under arbetet men en summering visar att 39 % läggs på systemutveckling och 52 % på systemförvaltningen. Avseende litteraturen fanns det 1999, när rapporten publicerades, 250 publikationer om systemutveckling och endast en om systemförvaltning.

Systemförvaltning och fortlöpande systemutveckling i en systemmiljö

Underhåll, dvs. systemförvaltning och utveckling är enligt Swanson (1999) nära sammankopplade. Swanson (1999) beskriver i sin artikel "IS "Maintainability": Should it reduce the maintenance effort?" att de båda involverar innovation av organisatoriska användare. Swanson (1999) beskriver vidare att systemförvaltning, som en "fortsatt utveckling", bakom korrigeringen, syftar till stor del att vara en del av den iterativa utvecklingen samt den pågående förbättringen av ett implementerat system. Det tillhandhåller användarna utökningar vilket ökar systemets värde och kan förlänga dess livslängd. Nyutveckling, i motsats siktar oftast på någon radikal förändring, där det befintliga systemet kommer att avvecklas och det nya systemet kommer att innebära förändringar för hur organisationen arbetar. Nya och framförallt ersättningsystem kan därför placera systemförvaltningen fel men även systemförvaltning kan göra så att organisationen kan förbise nyutveckling. Se figur 6 nedan.



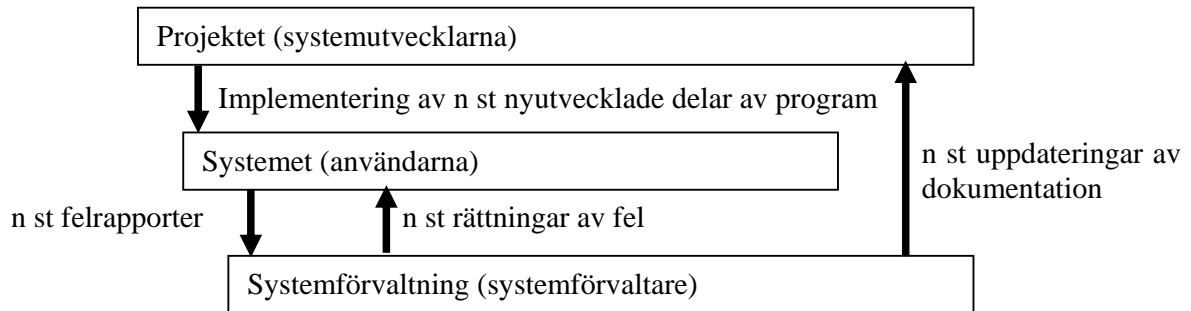
Notes:

- (i) Solid-line arrows indicate direct actions and effects, while dotted-line arrows indicate informative effects.
- (ii) The allocation of staff resources to new development and maintenance represents an informed choice and action, as is the expression of user needs in the form of new development and maintenance requests.
- (iii) Maintainability informs both choices and directly affects maintenance and its accomplishment, while transformability also informs both choices and directly affects new development and its accomplishment.
- (iv) The numbers refer to propositions associated with the direct actions and effects as they are discussed in the text.

Figur 6: Underhåll i sitt sammanhang. (Källa: Swanson, 1999)

Versionshantering

Som det beskrevs i introduktionskapitlet arbetar flera företag med utveckling av systemet samtidigt som en del av systemet används samt förvaltas. Användarna hittar genom att använda programmet ett antal fel som de rapporterar till systemförvaltningen. De rättar till felen och släpper vid olika tidpunkter nya versioner av programmet med rättningar. Systemutvecklarna jobbar samtidigt med fortsatt utveckling av programmet och släpper även de olika nya versionerna av programmet vid olika tidpunkter. Systemutvecklarna och systemförvaltarna jobbar med olika versioner som ska samköras innan de levereras till kunden. För att veta vilken version som är den mest aktuella använder de olika program (se kapitel "Verktyg för systemförvaltning"). Figur 7 nedan ger en bild över hur systemutveckling samt systemförvaltningen fortskrider.



Figur 7. Överskådlig bild problemområdet (Källa: Författarna (grundas på Révay, 1992)

ANALYS

I analyskapitlet redogör vi för de uppgifter som vi har fått fram genom våra intervjuer och diskuterar kring dessa utifrån det som vi har beskrivit i teorikapitlet, dvs. systemutveckling, systemförvaltning samt systemförvaltning samt fortlöpande systemutveckling.

Systemutveckling

Enligt Reváy (1992) utförs systemutveckling ofta i projektform. Reváy (1992) anser att för att kunna utföra ett effektivt projektarbete samt få ett effektivt styrinstrument bör systemutvecklingsmodellen väljas med noggrannhet.

Informant C framhäver vikten av användning av en formell och strukturerad systemutvecklingsmodell. Enlig informanten används alltid en mer eller mindre strukturerad systemutvecklingsmodell, i stora systemutvecklingsprojekt. Informant C hävdar att de alltid tillämpar en mer eller mindre iterativ systemutvecklingsmodell.

När man ser tillbaka på de stora systemimplementationerna som jag varit med om kan man konstatera att det alltid är en typ av iterativ systemutveckling – vare sig det var det som var planen eller inte. Det är väldigt sällan som man idag utvecklar ett affärssystem och sedan inte gör några ändringar i den levererad funktionalitet.

(Informant C)

Enligt informant C används någon typ av systemutvecklingsmodell vid all systemutveckling. Informant A anmärker att det är en tolkningsfråga vilken systemutvecklingsmodell de använder, men säger samtidigt att de i grunden jobbar med en iterativ utvecklingsmodell. Informant A gör dock en skillnad att jobba parallellt med iterationer respektive sekventiellt. Informanten hävdar att det kan ställa till med problem om iteration 2 påbörjas innan iteration 1 har avslutats.

Det är när du gör parallella iterationer dvs. att man börjar utveckling i iteration 2 på det som du tror du ska få fram av iteration 1 – som dom stora problem kommer.

(Informant A)

Den iterativa systemutvecklingen har inga bestämda regler eller ramverk vad gäller antal iterationer. Det finns ingen bra regel på vad en iteration ska innehålla (Sommerville, 2004). Antal iterationer styrs av typen av utvecklingsprojekt och komplexitet. Under ett halvt år säger informant B att de i snitt går igenom 10 iterationer i ett systemutvecklingsprojekt.

Om ett systemutvecklingsprojekt löper över 6 månader hinner vi gå igenom ca 10 iterationer. Men det är inget bestämt utan det varierar.

(Informant B)

Informant A poängterar att en iteration alltid måste resultera i något avgränsat och förutbestämt som går att leverera. Planeringen av en iteration, enligt informant A, grundar sig i planerat leveransdatum.

Vi räknar baklänges för att bestämma hur stor och hur många iterationer vi ska ha. Det bestäms av vad vi ska göra, hur mycket personal och andra resurser vi har att tillgå i projektet.

(Informant A)

Systemförvaltning

Begreppet systemförvaltning är väldigt brett och omfattar en mängd andra begrepp, så som software maintenance, systemunderhåll och viss mån vidareutveckling (Brandt 1992). IEEE definierar systemförvaltning

De totala antal aktiviteter vilka krävs för att ge kostnadseffektiv support till ett mjukvarusystem. (Blackburn et al. 2000).

Informant C definierar systemförvaltning som den totala mängden aktiviteter som tar vid när ett systemutvecklingsprojekt avslutas. Alla de tillfrågade informanterna har sin egen syn på systemförvaltningsbegreppet. Informant A säger:

Systemförvaltning handlar idag ofta om så mycket mer än bara hantera systemet/systemen och säkerställa att det fungerar som förväntat.

(Informant A)

Informant C utvecklar systemförvaltningsbegreppet genom att hänsyfta på att det även handlar om vidareutveckling samt felhantering.

Systemförvaltning är ofta lika mycket vidareutveckling som felhantering. Ofta så driver man igenom stora systemförändringar genom en systemförvaltnings organisation samtidigt som systemets existerande problem hanteras.

(Informant C)

Systemförvaltningsarbetet

Systemförvaltning initieras i ett avslutande skede av en utvecklingsprocess enligt den traditionella vattenfallsmodellen (Andersen, 1994). I en iterativ utvecklingsmodell, där varje delfas oftast är starkt begränsad och mindre än vid vattenfallsmodellen, initieras också systemförvaltningen i ett tidigare skede. Detta kan få positiva följd effekter eftersom det i ett tidigt skede ligger fokus på att göra systemförvaltningsfasen så lätt arbetad som möjligt (Shelly et al, 2002).

Nordström och Welander (2002) hävdar i sin bok Affärsmässig Förvaltningsstyrning att 33-75% av den totala kostnaden läggs på systemförvaltning. Enligt informant D's beräkningar läggs ca 80 % av den totala kostnaden för ett system på systemförvaltning. Detta är beräknat på ett systems totala livscykel. Informant C framhäver att systemförvaltningen är en väldigt betydande kostnad av ett systems totalkostnad under dess livscykel. Enligt informant C är systemförvaltningskostnaden en parameter som ständigt måste hållas under uppsikt för att inte överraska en organisation som implementerar ett nytt system.

Informant A säger att när de initierar nyutveckling av en mjukvara, ligger ingen fokus på systemförvaltning. Det finns enligt informanten A, i det skedet ingen personal som aktivt arbetar med systemförvaltning. Det finns några personer, 3-4 stycken, som inte allokeras för nästa systemutvecklingsprojekt för de är reserverade för systemförvaltningsarbetet. Efter första releasen (Appendix A) allokeras mer personal för att ta hand om systemförvaltningen.

När första releasen av systemet är färdig, sätts 10 – 15 % av den totala personal bemanningen under systemutvecklingsfasen av, för att arbeta med systemförvaltning. (Informant A)

Enligt informant C's organisations metod för systemutveckling tillsätts en person som ansvarig för systemförvaltningen i ett tidigt skede av systemutvecklingscykeln. Denna person har som ansvar att ta fram metoder och roller för systemförvaltningen. Personen spelar också en viktig roll som kvalitetssäkrare under projektets gång och kan därmed säkerställa att övergången till systemförvaltning blir så effektiv som möjligt, enligt informant C.

Révay (1992) rekommenderar att systemförvaltning bedrivs i en strukturerad projektform. Med projektet kommer många fördelar som underlättar för systemförvaltningen så som att projektet är avgränsat och tydligt definierat med en tydlig identitet och där det finns uttalade roller och ansvar. I enlighet med Révay (1992) beskriver informant D fördelarna med att bedriva systemförvaltning i projektform. Informanten poängterar vikten av att vid arbetet med systemförvaltning tydligt ha definierat och dokumenterat vad förvaltningsarbetet ska omfatta. Det viktigaste av alla aktiviteter är enligt informanten att det är förutbestämt vad som är att tolka som felrättning respektive vidareutveckling.

Det är vanligt att man i ett systemförvaltningsteam tvingas göra en massa vidareutveckling av systemet samtidigt som största fokus uttalat bör ligga på att rätta de existerande felen i systemen. Detta kan undvikas genom att det finns en tydlig avgränsning eller målsättning med vad förvaltningsteamet ska göra. (Informant D)

Enligt informant D's erfarenhet bedrivs idag mycket systemförvaltningsarbetet i någon form av projektform trots att där inte finns en tydlig avgränsning eller en tydlig start respektive mål. Informanten delar upp systemförvaltningen i två delar: Felhantering och vidareutveckling och gör följande rekommendation:

Rena fel som kräver enklare tekniska rättningar eller processförändringar bör hanteras av en del av systemförvaltnings-teamet som kan vara linjeorganiserat. Vidareutveckling dvs. förändringar eller ny funktionalitet i systemet bör hanteras av resurser som är organiserade i projektform. På detta sätt får man det bästa av två världar. (Informant D)

Révay (1992) har fastsällt att en av faktorerna för att framgångsrikt kunna få till stånd en förändring på synen på systemförvaltning, från linje organisatorisk till framgångsrik projektorienterad systemförvaltning, är serviceandan i systemförvaltningsverksamheten. Informant C framhäver vikten av att personer som arbetar med förvaltning är de duktigaste och mest motiverade som finns att tillgå. Informant C påpekar att det är väldigt viktigt att rätt personer som arbetar med systemförvaltning, i synnerhet när det handlar om affärssystem.

Det måste finnas en medvetenhet bland de som arbetar med förvaltning om hur viktigt det är för verksamheten att systemförvaltningen fungerar. Uppstår det några problem med systemet måste alla användare och intressenter känna tillit till att systemförvaltningen kommer att fixa problem. (Informant C)

Ändringshantering

Förvaltningsorganisationen som bildas efter första releasen (Appendix A) inom informants A's företag kan utföra tre olika uppgifter. Den första innebär att förvaltningsorganisationen mottar felrapporter och begäran om förändringar och prioriterar dessa på en skala. Den andra är att Förvaltningsorganisationen hanterar felrapporten/förändringsbegäran om de har möjlighet (resurser, befogenhet, kompetens) Den tredje uppgiften är att fördela felrapporten/förändringsbegäran om de inte har möjlighet att hantera den. I dessa fall lämnas den över till lämplig så kallad design enhet, som har resurser avsatta för att hantera dessa.

Informant D beskriver de 4 viktigaste uppgifterna för en förvaltningsgruppchefsroll vid direkt arbete med felrapporter eller förändringsbegäran enligt följande. 1 – Godkänna eller avslå felrapporten alternativt förändringsbegäran. 2 – Prioritera. Alla felrapporter och förändringsbegäran rapporteras av användare i ett Web-baserat verktyg. Dessa prioriteras på en skala från 1 till 5 där 1 är den högsta prioritet. 3 – Tidsuppskattning. Varje felrapport eller förändring måste tidsuppskattas för att skapa kontroll över tillgängliga resurser. 4 – Distribuera till rätt resurs (person) för att optimera resursanvändandet.

Att prioritera inkommande fel är ett omfattande arbete enligt informant A.

Från början hade vi tre olika prioriteringskategorier, A, B, C. Nu har det tillkommit ca 20 stycken olika prioriteringskategorier till, exempelvis "Hot", eller "Hot Hot". (Informant A)

Prioriteringar av felrapporter styr implementeringen av felrättningen den så kallade fixen (Appendix A) enligt informant C. Vidare anser informanten att det är lätt att det går inflation i prioriteringar och att lösningen ofta blir att nya högre prioriteringsbegrepp läggs till. En högt prioriterad fix implementeras som en så kallad Hot Fix, det innebär enligt informant C att fixen läggs in i systemmiljön (Appendix A) omedelbart, det är alltså en fix som har betydande relevans för att systemet ska fungera ändamålsenligt. Informanten påpekar i detta sammanhang att det föreligger en stor risk för fel vid implementering av Hot Fixar.

Det är enormt viktigt att förändringen läggs in i alla aktiva systemmiljöer och att versionshanteringsprogram uppdateras med den senaste fixen, att all dokumentation uppdateras och att alla involverade parter informeras. (Informant C)

Definition av felrapporterna är något som informant D anser vara av stor betydelse. Informanten framhäver vikten av att inte felrapporter och förändringsbegäran blandas och fördelas till samma resurser.

Det är viktigt att den personen som tar emot och distribuerar felrapporterna vet vad det handlar om och att inte en systemförändring kommer in och rapporteras som ett fel – om så inte är fallet. Det handlar om att ha lite fingertoppskänsla och en hel del systemkunnande både tekniskt och funktionellt. (Informant D)

Problem med underhåll av programvaror

Eklund & Fernlund (1998) hävdar att en av orsakerna till att det görs inskränkningar målsättningarna att skapa en strukturerad och underhållsvänlig programvara är att förvaltning har relativt låg status hos programleverantörerna. Informant C gör motsvarande konstaterande som Eklund & Fernlund (1998) och tillägger även att förvaltning ofta har låg status i allmänhet.

Förvaltnings arbete ses ofta som ett mindre kreativt arbete – mer som ett städ jobb istället för ett skapande arbete. Det är ofta jag får uppfattningen att förvaltningsarbete ses som mindre krävande och utmanande än nyutveckling – Det är i grunden en helt felaktig syn. Förvaltnings arbete är ofta mera komplext och krävande än vad nyutveckling är (Informant C)

Informant D kritiserar de systemleverantörer (Appendix A) av standardsystem som informanten har erfarenhet att arbeta med när det gäller att vidta pro-aktiva åtgärder vid utvecklingen för att underlätta underhåll av systemen. Informanten hävdar att ofta både kod och dokumentation är av ganska låg kvalitet och att standardsystemleverantörerna fokuserar för mycket på funktionalitet och för lite på underhållvänlighet. I enlighet med Eklund & Fernlund (1998) ovan, säger informanten att förvaltning är något som har låg status och ofta också låg prioritet hos systemleverantörerna.

Eklund & Fernlund (1998) har identifierat tre kategorier där det kan förkomma sideeffekter vid systemunderhåll: Kod, Data och Dokumentation. Informant D framhäver den komplexitet som de flesta stora affärssystem innefattar. Informanten identifierar kod förändringar som det absolut mest komplexa.

Det är relativt enkelt att göra en kod förändring i enlighet med en felrapport, men det är ofta väldigt komplext att genomföra en fullständigt korrekt påverkansanalys. Det händer ofta att en förändring rättning leder till följdfel som först upptäcks när förändringen är implementerad (Informant D)

Enligt Eklund & Fernlund (1998) är Data en av de kategorier som kan leda till sideeffekter. Det är något som Informant C bekräftar. Informanten poängterar problematiken med databas baserade system där systemförvaltningsteamet gör manuella uppdateringar i databasen utan att ha full förståelse för vad en uppdatering kan få för konsekvenser.

Dokumentation

Enligt Eklund & Fernlund (1998) bör existerande dokumentation av programvara uppdateras vid en förändring av systemet. Författarna rekommenderar att det utförs en detaljerad underhållsrapport.

Informant D ser underhåll av system dokumentation som ett område som frekvent förbises vid systemförvaltning.

Jag tror att den största orsaken är tidspress och avsaknad av motivation. För mindre ändringar kan det ofta vara så att uppdateringen av dokumentationen tar längre tid än själva fixen. (Informant D)

Vidare påpekar informant D att ofta så används felrapporteringsverktygen för att dokumentera vilka förändringar som görs. Informanten beskriver en dokumentation i felrapporteringsverktygen som är väldigt lik den underhållsrapport som Eklund & Fernlund (1998) rekommenderar.

Verktyg för systemförvaltning

Versionshanterings program är en av de viktigaste redskapen vid systemförvaltning enligt informant D. Informanten framhäver vikten av att alla personer inblandade använder och förstår vikten med versionshanteringen. Versionshanterings verktyg är en lika viktig beståndsdel som kvalitativ dokumentation enligt informant C.

Utan ett fungerande och korrekt använt versionshanteringssystem är det så gott som omöjligt att bedriva fungerande systemförvaltning. Detta gäller speciellt när det bedrivs vidareutveckling på samma system. (Informant C)

Informant D har erfarenhet av flertalet olika versionshanterings program och felhanteringsverktyg och gör ingen större skillnad på de olika verktygen. Informanten understryker vikten av att verktyget används:

Den viktigaste faktorn vid val av versionshanterings och felhanteringsverktyg bör vara att personerna som ska använda dem kan acceptera dem och att de passar eller är anpassningsbara för det de ska användas till.

(Informant D)

Systemförvaltning och fortlöpande systemutveckling i en systemmiljö

Enligt informant D arbetar de oftast med affärssystem. De arbetar då i en miljö där det förkommer systemförvaltning och systemutveckling parallellt. Informant D anser att det är en definitionsfråga om vad som klassificeras som systemförvaltning respektive systemutveckling. Vid systemförvaltning av ett modernt affärssystem hävdar informant D att systemförvaltningen oftast går att ses som systemutveckling på grund av de ofta komplexa förändringar/rätningar som initieras och drivs av systemförvaltningsteamet. Informant D betonar, så som övriga informanter, komplexiteten och de omfattande felkällorna som finns när systemförvaltning och systemutveckling bedrivs inom samma system.

Ibland är det svårt att klassificera vad som är systemförvaltning och vad som är systemutveckling eftersom det i vissa delar av projektet är samma person som dels utvecklar men även rättar fel.

(Informant D)

Informant A är verksam i en organisation där systemutvecklingsprojekten är uppdelade i flera olika iterationer, där varje iteration ska resultera i något levererbart. Varje iteration är sedan indelade i mindre delar som fungerar som kontroller. De har en processbeskrivning som beskriver hur de ska ta hand om felrapporter.

Processbeskrivningen bestämmer både pro-aktivt och re-aktivt hur vi ska arbeta.

(Informant A)

När en felrapport resulterat i en felrättning måste den felrättningen kopieras till de olika systemmiljöerna. Enligt informant B är det kopieringen av felrättningen som är den tydligast största källan till att det blir problem när systemförvaltning bedrivs samtidigt som systemutvecklingen fortlöper. Även informant D betonar att systemförvaltning av ett system som det samtidigt bedrivs omfattande vidarealternativt nyutveckling av innebär stora risker vad gäller hantering av olika versioner i olika miljöer.

Att lägga till ett fel som någon har skickat in måste göras noggrant, dvs. vi måste se till att det läggs till i de olika systemmiljöerna, annars kommer vi kanske att få något nytt fel någon annanstans.

(Informant D)

I den organisation som informant A jobbar inom bedrivs utvecklingen i två parallella systemutvecklingsprojekt och en förvaltningsorganisation. Informant A påpekar att det inte är optimalt att parallellt bedriva systemförvaltning och systemutveckling, men säger samtidigt att de är nödvändigt för att kunna leverera produkter i den takt som marknaden kräver. Informant C har motsvarande förklaring och framhäver också att det blir allt mera vanligt förekommande att det bedrivs systemförvaltning och systemutveckling parallellt inom många organisationer.

Vill vi vara med i toppen måste vi kunna leverera produkter till kunder och då måste vi ha korta utvecklingstider, annars tar någon annan jobben.
(Informant A)

Informant C påpekar kravet på snabba synliga resultat, från både beställare och användare (Appendix A). Detta leder ofta till att det ligger fokus på att implementera de delar av ett system som kan ge en direkt ekonomisk avkastning. När dessa delar är implementerade och har ingått i förvaltningsfasen av livscykeln påbörjas utveckling av näst version.

Kunden vill oftast se resultat direkt - så fort ett systemutvecklingsprojekt har initieras - detta ställer stora krav på att snabbt kunna ge kunden något att använda!
(Informant C)

Enligt Swanson (1999) beskriver korrigeringar av ett system, som sker under systemförvaltning som "fortsatt utveckling" vilket innebär att inte enbart rena rättningar görs utan även förbättringar och funktionell utveckling av ett system. Enligt Swanson (1999) är detta en naturlig del av en iterativ utvecklingscykel.

Informant C har identifierat ett par huvudområden som det enligt informanten bör läggas fokus på när det bedrivs förvaltning och systemutveckling parallellt:

1. Tydligt ägandeskap vad gäller olika systemmiljöer. Enligt informant C är det av vikt att det finns en eller flera personer som är ytterst ansvariga för en systemmiljö och har en övergripande kontroll på vilka versioner och förändringar som är implementerade.
2. Förändringar av dokumentationen i relation till faktiska systemförändringar bör enligt informant C prioriteras lika högt som själva systemförändringen. Enligt informant C bör fokus vara lika mycket på att distribuera och förmedla den senaste versionen av dokumentationen som förmedlingen av systemförändringen.
3. Ett fungerande och korrekt använt versionshanteringsprogram (mjukvara) är enligt informant C en förutsättning för att kunna hantera denna parallella problematik.
4. Att tillgängliga verktyg och process stöd används och att det sker uppföljning på att det används.

Informant D refererar ofta till kravspecifikation (Appendix A), dvs. kravet från kunden, på att snabbt leverera förändringar/förbättringar av systemen och hur dessa krav från kunden ofta leder till en stressfull arbetsmiljö där systemutvecklare (Appendix A) och systemförvaltare (Appendix A) slarvar med att följa givna processteg.

Det handlar för det mesta om en ganska svår avvägning. På ena sidan har du kunden som vill kunna utnyttja den nya funktionaliteten så fort som möjligt. På andra sidan finns kraven på att leverera hög kvalitet utan buggar och följa de definierade processteg som finns, för att undvika eventuella problem. När kundens önskemål på snabbhet tar överhanden – det är oftast då det uppstår problem. Man gör helt enkelt mer eller mindre omedvetna avkall på processtegen.
(Informant D)

Informant A tar upp liknande problem med att bedriva två parallella systemutvecklingsprojekt kan leda till att det blir en stressig miljö som kan skapa problem pga. av mänskliga misstag.

Ibland när det är stressigt och tiden är knapp är det ofta en ökad risk för mänskliga misstag, i samband med distribution och uppdatering av olika systemversioner.
(Informant A)

Att ha spårbarhet till kravspecifikationen är något som informant A anser vara viktigt. Informant B framhäver även vikten av hela tiden kunna se vilka av kraven som implementerade, verifierade, pågående osv. Informant B framhåller även vikten av att kunna visualisera den information som idag finns inom organisationen för projektledarna och styrgrupperna. Att få information om ett projekts status i siffror är ibland inte tillräckligt. Det kan även vara så att för mycket siffror gör det svåröverskådligt. Informant B arbetar bl a med att visualisera den information som finns runtomkring projektet. Ett exempel på den visualiseringen är antalet krav som är implementerade, verifierade eller testade. Utifrån den informationen kan sedan projektledarna stämma av ett projekts status.

Är ett projekt efter kan t ex mer folk eller andra resurser sättas in för att stödja projektet.
(Informant B)

Rättningspaket kommer ut med viss regelbundenhet. Enligt informant A är det en högre frekvens i början, dvs. efter första releasen (Appendix A), i jämförelse med senare. En anledning är att behovet är större i början. Är det fel som är mycket akut så rättas den direkt och skickas till kunden. En bedömning görs sedan om rättningen även ska ingå i nästa rättningspaket som alla kunder tar del av. En del rättningar är enbart för vissa kunder eftersom de har olika användningsområden för programvaran. Enligt informant A finns det ingen bestämd storlek på rättningspaketen.

En uppskattning är att de har ca 100-150 rättningar per rättningspaket.
(Informant A)

Ett problem som informant A upplever är att få kunderna att integrera de nya rättningspaketen i deras befintliga programvara. När de har ett system som fungerar ser de inte vinsten med uppdateringen. Enligt informant A jobbar många företag utifrån filosofin:

Vi vet vad vi har men vi vet inte vad vi får.
(Informant A)

Enligt informant C är det en av de grundläggande faktorerna som skapar problem att det förekommer så många intressenter i ett system. Informanten ger exempel på systemmiljöer där det finns en standardsystem leverantör, ett systemutvecklingsteam och ett systemförvaltningsteam. Samtliga tre enheter jobbar med förändringar/förbättringar av systemet. Standardsystem leverantören jobbar oftast med felrättningar i standard versionen, medan de övriga två teamen jobbar med anpassningar och felrättningar som bygger på grundläggande anpassningar som gjorts av standardsystemet för att anpassa det efter kundens önskemål. Enligt informant C's erfarenhet är den största felkällan att inte systemleverantörens (Appendix A) förändringar implementeras i systemen. Ett problem som informant C nämner är vissa uppdateringar, även kallade 'patchar' (Appendix A), implementeras medan andra inte gör det.

I många fall när man anpassar ett system åt en verksamhet så tar man inte tillräckligt med höjd för att vara förberedd på att systemleverantören ska komma med stora förändringar. Man bygger anpassningarna på de förutsättningar som finns i systemet när det först levereras. Detta innebär att när det kommer en 'patch' från systemleverantören så är det inte alltid man väljer att implementera denna. Om det sen kommer en ny 'patch' som man vill implementera och väljer att implementera denna finns det stor risk att föregående 'patch' var en förutsättning för att den nya 'patchen' ska fungera. Detta kan förstås få ödesdigra konsekvenser. Det är väldigt komplext och svårt att ta korrekta beslut.

(Informant C)

Enligt informant B är det en stor skillnad i vilka typer av fel som rapporteras samt vad de orsakas av. I början beror ca 90 % på att de inte vet vad de ska göra men senare i projektet är det kanske bara ca 20 %. Informant A framhäver ett annat problem.

Den största procenten av fel som vi får är fel som vi har rättat men som vi har haft lite bråttom med, t ex att vi inte har kollat hur en rättning av ett fel påverkar andra delar av systemet.

(Informant A)

Ett stort problem som informant A framhäver är kontrollen av vilka kunder som har fått vilka rättningar/rättningspaket samt vilka versioner av rättningspaketen de har. När en rättning som är akut har skickats ut till kunden är det viktigt att veta vad det är som har rättats samt att det är testat i den befintliga systemmiljön. Informant A berättar om ett exempel, som händer enskilda gånger och då oftast tidigt i livscykeln, när en viktig kund har rapporterat ett fel som har rättats. En tid senare skickas sedan ut ett rättningspaket till kunden vilket leder till att felet återkommer.

Då blir kunderna mycket missnöjda. Det är inget annat än att vi inte har bra kontroll.

(Informant A)

Kontroll av vad som har rättats samt vad det är som har utvecklats i vilken utvecklingsmiljö är även något som informant B framhåller som ett stort problem. De har idag programvara för kontroll av vilka versioner som kunden/kunderna har som de anser fungerar. Informant B anser att programvaran är tillräcklig för de behoven som de har men framhåller att det största problemet är den manuella hanteringen.

Den programvara vi använder idag funkar, men många vet inte att använda den rätt. Skulle alla personer göra det skulle många av våra problem försvinna, (Informant B)

Versionshantering

Informant D framhäver vikten av att använda ett versionshanteringsprogram som är accepterat av alla intressenter och som alla klarar, kan och vill använda. Användarna måste förstå orsak och verkan av att på ett korrekt sätt använda existerande versionshanteringsprogram.

Det spelar ingen roll om man har världens bästa versionshanteringsverktyg och processer om inte användarna på ett korrekt sätt använder verktyget. (Informant D)

I den organisation som informant B arbetar i kopieras felen till olika utvecklingsmiljöer. Felet rättas sedan vanligtvis av en person i de olika utvecklingsmiljöerna. Informant B anser att den processen inte är problematisk men anser att det är steget därefter som är kritiskt. Det steget informant B avser är när den mänskliga faktorn arbetar med versionshanteringssystemet. Blir det fel någonstans leder det ofta till andra fel.

Det räcker inte med ett jättebra versionshanteringssystem när det finns möjlighet till misstag av användaren. (Informant B)

SLUTSATSER

Det första syftet med uppsatsen var att beskriva hur systemförvaltning samt fortlöpande systemutveckling bedrivs i en systemmiljö. Den teori som finns tillgänglig inom ämnet är begränsad. Avsikten med undersökningens första syfte har tillgodosetts i första avsnittet av teori- och analyskapitlet. I teorikapitlet beskrivs systemutveckling som en process som pågår till den tidpunkt då leverans av programvaran sker. Systemförvaltningsarbetet beskrivs som den process som tar vid när programvaran har levererats. Arbetet med systemförvaltning och fortlöpande systemutveckling är ett arbetssätt som används inom flera organisationer. I analyskapitlet beskrivs den syn de olika informanterna har på systemutvecklings- och systemförvaltningsarbetet. Arbetssättet innebär att viss funktionalitet alternativt en viss del av programvaran utvecklas och levereras till kunden vid olika tidpunkter. Detta innebär att kunden snabbare kan börja använda den beställda mjukvaran och realisera de vinster som den förhoppningsvis medför. När kunden/användarna börjar använda programvaran identifieras ofta fel (buggar) som rapporteras till systemförvaltningsteamet som i sin tur rättar felen samtidigt som systemutvecklingsteamet fortsätter att utveckla programvaran. Informanterna påpekar att arbetssättet är komplext om än ofrånkomligt och något som blir allt mer vanligt förkommande. Huvudanledningen till det förändrade arbetssättet är för att tillgodose kundernas krav på att snabbt kunna se vinster med den beställda programvaran. Detta bör jämföras med att utveckla och leverera hela programvaran vid en tidpunkt. I teorikapitlet beskrivs de olika verktyg som de olika informanterna har erfarenhet från och som deras respektive organisationer/projekt använder vid systemförvaltningsarbetet.

Det andra syftet var att finna och beskriva de framgångsfaktorer som bör tas i beaktande vid systemförvaltning och fortlöpande systemutveckling. Samtliga informanter beskriver systemförvaltning och fortlöpande systemutveckling som problematiskt och svårhanterbart. Detta framkommer i den andra delen av analyskapitlet.

Följande faktorer har i analysen framkommit som tongivande för att framgångsrikt bedriva systemförvaltning och fortlöpande systemutveckling:

- Valfungerande versionshantering
- Uppdaterad systemdokumentation
- Definiera systemförvaltning respektive systemutveckling
- Ägandeskap av system/systemmiljöer
- Processbeskrivning för systemförvaltningsarbetet
- Konfirmering av installation av programuppdateringar

Det mest signifikanta problemområdet som framkommit under analysen är versionshantering och områden som relaterar till versionshantering. Problemet beskrivs oftast uppstå när ett systemförvaltningsteam och ett systemutvecklingsteam arbetar med olika versioner och med olika typer av förändringar som relaterar till varandra. När dessa versioner sedan ska integreras till en gemensam version alternativt distribueras till en eller flera miljöer/kunder föreligger en stor risk för att diskrepans mellan de olika miljöerna uppstår. Allt versionsrelaterat arbete kräver stor noggrannhet från de inblandande parterna, framförallt systemförvaltningsteamet och systemutvecklingsteamet. Tillgängliga verktyg och versionshanteringsprocesser måste vara accepterade av samtliga intressenter. Det är också av största vikt av att alla intressenter använder befintliga verktyg på ett korrekt sätt.

Förändringar av systemdokumentationen i relation till faktiska systemförändringar prioriteras inte lika högt som själva systemförändringen. Detta är ett problem vilket kan få följd effekter i form av att framtida förändringar eller systemanalyser baseras på felaktig och obsolet information. För att undvika problem bör det ligga lika mycket fokus på att uppdatera och distribuera den senaste versionen av dokumentationen som att distribuera en systemförändring.

Utifrån informanternas erfarenhet har det framkommit att definitionen av systemförvaltning respektive systemutveckling kan spela en betydande roll. Enligt undersökningen är begreppet systemförvaltning delvis synonymt med uppstädning/rättning av kod. Det kan få konsekvensen att det blir svårt att få tag i kompetenta och motiverade resurser. En lösning för att kringgå detta är att inte enbart definiera systemförvaltning som uppstädning eller rättning utan att definiera och associera systemförvaltning med systemutveckling. En anledning är att höja den upplevda statusen på det arbete som utförs inom systemförvaltningen.

Ett problem som framkommit är ägandeskapet vad gäller olika system. Det är viktigt att det finns en eller flera personer inom en organisation som är ytterst ansvariga för ett system, dvs. systemägare (Appendix A). Ägandeskapet bör omfatta all information om det enskilda systemet, hur det används samt hur det integrerar med andra system. Ägandeskapet bör också innefatta ett ansvar för att ha en övergripande kontroll på vilka versioner och förändringar som är implementerade.

Den mänskliga faktorn är att beakta som något att alltid räkna med som en potentiell felkälla. Detta är något som samtliga informanter har lyft fram i undersökningen. Det som avses med den mänskliga faktorn är i det här fallet inte samarbetsproblem utan att personer exempelvis använder programvaror eller processer på ett felaktigt sätt. Detta kan i sin tur leda till fel som är svåra att upptäcka, t ex gammal dokumentation eller att fel som är rättade inte är implementerade i alla miljöer trots att de finns väldokumenterade. Det mest effektiva sätt att minimera riskerna med den mänskliga faktorn är att ha väldokumenterade och välgrundade processbeskrivningar som samtliga involverade partner accepterar.

Den sista faktorn som framkommit i undersökningen relaterar till ett scenario där det förekommer en kund - leverantör relation. Det har framkommit att det finns ett visst motstånd till att implementera uppdateringar av system som inte direkt går att relatera till någon förbättring av ett systems existerande funktionalitet. Kunderna är restriktiva för programuppdateringar då de är nöjda med den version de använder. Problemet

föreligger då inte leverantören är införstådd med vilken version som kunden har implementerat. Detta kan leda till okontrollerbara effekter om leverantören levererar en ny version som inte är baserad på den version som kunden använder. Det är därför av stor vikt för en leverantör att kunden alltid konfirmerar att de har implementerat den senaste versionen.

Framtida forskningsfrågor

Uppsatsen har resulterat i ett antal framgångsfaktorer som bör tas i beaktande vid systemförvaltning och fortlöpande systemutveckling. Det finns ett antal frågor som skulle vara intressant att forska vidare om. Det skulle vara intressant att studera en organisation som tar hänsyn till dessa framgångsfaktorer.

Ett annat forskningsområde som skulle vara intressant att studera är påverkan av framgångsfaktorerna ur olika synvinklar, t ex systemförvaltare, projektledare, systemutvecklare, supportpersonal osv.

KÄLLFÖRTECKNING

- Arbnor, I. & Bjerke, B. (1994) *Företagsekonomisk metodlära*, Studentlitteratur, Lund
- Andersen, E. (1994). *Systemutveckling: principer, metoder och tekniker*. Lund: Studentlitteratur.
- Backman, J. (1998). *Rapporter och uppsatser*. Lund: Studentlitteratur.
- Belina, F., Hogrefe, D. (1989). *The CCITT-specification and description language SDL*. Computer Networks and ISDN Systems, 16,
- Blackburn, J., Scudder, G. och Van Wassenhove, L. N. (2000). *Concurrent software development*. ACM.
- Brandt, P (1992). *Hur bedriver man systemförvaltning*. Stockholms Universitet.
- Brandt, P., Carlsson, R., Nilsson, A. G. (1998). *Välja och förvalta standardsystem*. Lund: Studentlitteratur.
- Budgen, D. (2003). *Software design*. England: Addison-Wesley
- Ejvegård, R. (1993). *Vetenskaplig metod*. Lund: Studentlitteratur.
- Eckholm, M. & Fransson, A. (1975) *Praktisk intervjuteknik*, Almqvist & Wiksell Förlag AB, Malmö
- Eklund, S. och Fernlund, H. (1998). *Programkonstruktion med kvalitet – projekthantering och ISO 9000*. Studentlitteratur, Lund.
- Eriksson, L.T. och Wiedersheim-Paul, F. (1999). *Att utreda forska och rapportera*. Malmö: Lieber AB
- Eriksson, L.T. och Wiedersheim-Paul, F. (1997). *Att utreda forska och rapportera*. Malmö: Lieber Hermods AB.
- Haverblad, A. (2004). *IT service management i praktiken*. Lund: Studentlitteratur.
- Holme, I. och Solvang, B. (1997). *Forskningsmetodik – Om kvalitativa och kvantitativa metoder*. Lund: Studentlitteratur.
- Nordström, M., Welander, T. (2002). *Affärsmässig förvaltningsstyrning*. Lund: Studentlitteratur.
- Patel, R. och Davidsson, B. (1991). *Forskningsmetodikens grunder*. Lund: Studentlitteratur.

Pressman, R.S. (2000). *Software Engineering, a practioner's approach*. Cornwall: McGraw-Hill.

Révay, P. (1992). *Modern systemförvaltning*. Lund: Studentlitteratur.

Ross, D. (1985). *Application and extension of SADT*. Computer 18, 4,

Shelly, G.B., Cashmant, J., Rosenblatth, J. (2002). *Systems Analysis and Design*. Thomson Learning, Canada.

Sommerville, I. (2004). *Software Engineering*. USA: Addison Wesley.

Stewart, W. (1999) *Interviewing: Principles and Practices*, McGraw Hill College, Boston

Swanson, E. B. (1999). *IS Maintanability: Should it reduce the maintenance effort?*. ACM

Thurén, T. (1998) *Vetenskapsteori för nybörjare*, Liber AB, Stockholm

Wallén, G. (1996). *Vetenskapsteori och forskningsmetodik*. Lund: Studentlitteratur.

Yin, R. (1994) *Case Study Research: Design and Methods*, Sage, Thousand Oaks

Källor internet

ACM <www.acm.org>. Avläst 040923.

Computer Associates <<http://www3.ca.com/Solutions/Product.asp?ID=255>>. Avläst 041101.

IBM Rational (1) <<http://www-306.ibm.com/software/awdtools/clearcase/>>. Avläst 041101.

IBM Rational (2) <<http://www-306.ibm.com/software/awdtools/clearcase/>>. Avläst 041101.

IEEE <www.IEEE.org>. Avläst 040924.

Microsoft, MSDN <<http://msdn.microsoft.com/vstudio/previous/ssafe/>>. Avläst 041101

APPENDIX

A. Grundläggande begrepp

Användare

Med användare avses i förstahand slutanvändarna det vill säga de som arbetar med front-end delen av ett system.

Beställare

Beställare kan ofta likställas med en kund som på högsta nivå oftast är en organisation. Beställaren kan även brytas ner till lägre nivåer – vanligtvis kan en CIO vara den övergripande beställaren och diverse funktionella alternativ avdelningschefer vara beställare.

Fix

En rättning av ett fel i koden.

Kravställare

Kravställare kan vara alla intressenter som ska använda ett system eller som har ansvaret för ett system. En kravställare har befogenheter att ställa krav på hur ett system ska fungera eller se ut – beroende på typ av system.

Moduler

Med moduler avses de delar eller enheter som ingår i en programvara.

Kravspecifikation

Ett dokument som specificerar krav på ett informationssystem eller dess delar.

Patch

En eller flera filer som innehåller uppdateringar av programvaran.

Prototyp

En preliminär typ, form eller instans av ett system som stödjer en modell för senare utvecklingssteg eller slutlig version av systemet.

Release

Distribution av en ny produkt eller en ny funktion och rättelser i en redan befintlig produkt

Systemförvaltare

Den som i huvudsak administrerar förändringsarbetet

Systemleverantör

Med systemleverantör avses de företag som leverera standardiserade system. Dagens systemleverantörer består idag av till stor del större företag med flertalet kunder inom samma eller motsvarande bransch.

Systemutvecklare

Arbetar med utveckling av ett system.

Systemutvecklingsprojekt

Utveckling av programvara sker vanligtvis i projekt (Eklund & Fernlund, 1998). Olika människor har olika uppfattningar om projekt men det finns sju punkter som kan ge en övergripande beskrivning av vad ett projekt är (Eklund & Fernlund, 1998):

Ett projekt är av engångskaraktär

Ett projekt är målinriktat

Ett projekt är avgränsat i tid och omfattning

Ett projekt är uppbyggt kring en tillfällig organisation

Ett projekt är tilldelat begränsade resurser

Ett projekt är planerat med avseende på tid, resurser och kostnader

Ett projekt är uppdelat i delmål

Systemägare

Den som har det yttersta och övergripande ansvaret för informationssystem

Systemmiljö

Med systemmiljö avses de hård- och mjukvaror som finns i given organisation. En systemmiljö kan vara en del av en större systemmiljö och då omfatta mjukvara och hårdvara som används vid exempelvis vid testing, utveckling eller i produktionsmiljön.

Test

En aktivitet varvid ett programsystem eller en del av ett programsystem exekveras som en kontroll av att programsystemet eller delen uppfyller ställda krav

Ändringsbehov

Är en generisk benämning för alla slags krav, rapporter, dokument (t ex felrapport, systemändring, ändringsbegäran, ändringskrav) som avser samtliga förändringar av informationssystem.

B. Intervjuguide

Bakgrund till utvecklingsprojektet?

Hur ser systemmiljön ut?

Vad/Hur påverkar de olika aktörerna systemmiljön?

Går det att beskriva hur det hänger ihop?

Släpper ni i förvaltningsteamet egna version av systemet?

Beskriv hur processen ser ut för att införa en ny version av ett system/delsystem. Från det att det finns en version som ska släppas till det att den är implementerad?

Gör utvecklaren själv förändringen i systemet?

Hur fungerar det med att samla ihop förändringar?

Arbetar ni utifrån systemutvecklingsmodeller?

Hur kommunicerar ni med utvecklingsteamet och hur ser ni till att ni alltid har samma version?

Förebygger ni problematiken med olika versioner med hjälp av något program?

Vad gör ni för pro-aktiva åtgärder för att minska risken att fel version hamnar i miljön?

Hur är fördelningen av systemförvaltningsarbetet?